A Discussion of Internet Research Focusing on Mobility and Next Generation Protocol Development

Wayne Mackrell
Department of Computer Science
University of Saskatchewan
57 Campus Drive
Saskatoon, SK S7N 5A9
CANADA
e-mail: wlm125@cs.usask.ca

Supervisor: Rick Bunt

November 28, 1996

Contents

1 Introduction

In the last two decades, the use of computers has grown dramatically. Computers are now very prominent in most vocations and have even invaded our homes. Currently computers are used by people of all ages (except maybe the youngest of infants?). Uses include games, financial planning, word processing, complex simulation, computer assisted design, computer assisted engineering, and many others.

The newest capability being exploited in the home and office is internetworking with other computers both within an organization and around the world. Internetworking at such a large scale is possible today because of the *Internet*. Many of us are familiar with the Internet at some level. This document will provide some background information on the Internet and will discuss research in the Internet as it applies to mobility support and next generation protocol development. Although a general introduction to Internet research will be given, the primary focus of this paper is on *Mobile IP* which is an extension to Internet protocols that facilitates the mobility of computers on the Internet. Current research in the Internet also involves the generation of a new Internet protocol call *IP version 6*. This paper will introduce IP version 6 by discussing some of its enhancements over IPv4. Of particular interest to this paper are the facilities being built into IP version 6 to support user mobility (see Section 11).

2 An Introduction to the Internet

At the heart of the Internet are the computers and networks of which it is made. The computers on the network are commonly called *hosts* or *nodes*. In this paper we will refer to computing devices using the Internet as hosts. *Networks* are collections of hosts than can communicate with each other over some shared medium. Individual businesses, organizations, and institutions will commonly have individual networks that their personnel use. The Internet is an example of a larger entity called an internetwork. Internetworks are collections of networks which allow hosts to communicate across network boundaries according to sets of rules called *internetwork protocols*. The Internet itself is the largest internetwork in the world, and connects many individual networks (in the tens of thousands). Many users are familiar with the Internet today and its popularity has been greatly increased by the development and exposure of the *World Wide Web*.

The Internet itself can be thought of from many different perspectives, and appears as a different entity from each perspective. To many users the Internet appears as a large collection of data around the world that can be accessed using various network programs. The most common Internet exploration programs being used today are World Wide Web (WWW) browsers which present the Internet as a large collection of colourful and informative pages linked to each other by special portions of the page which can be selected with a pointing device such as a mouse. To some people, the Internet is a global talk session where you can interact in real-time with people all around the world. Some people use the Internet for wide-area videoconferencing with many people. Others use the Internet to read news articles or to send messages (e-mail) to one another. A lower level, although very useful, view of the Internet is as a large collection of computers which allow remote access. All of these views are valid interpretations of the Internet, and in fact, the Internet is all these things.

The Internet is now a worldwide communication medium which is being used by an ever-increasing number of users for a growing number of activities. Originally, the Internet was designed in a military context for connecting different military networks. Eventually, the users of the Internet grew to include educational institutions. The inclusion of educational institutions greatly affected the Internet because of the increased research capacity available [BTB95]. Eventually, particularly in the 1990's, the use of the Internet spread to the general public through various network providers.

The inclusion of the general public, and the development of the World Wide Web [BLCGP92], has greatly increased traffic and the demand on Internet resources. In recent years, the Web has been the fastest growing component of network traffic on the NSFNET backbone (the major packet transmission lines in North America) and throughout the Internet [AW96]. Even though the NSFNET has now been phased out, all indications are that the statistics apply equally well in the current architecture of NAPs (Network Access Points)[NSF95] which are controlled by Internet service providers. The Web has been around only a few years, but its development has exposed many people to the Internet. The Internet itself and its various problems, however, have been around a lot longer.

The development of the Internet was a major leap in communications technology. Early technology supported network-wide connectivity, but these methods of communicating (characterized by the protocols being used) did not support connections between networks. Interconnection was needed by the military to exchange data between its networks. Various interim protocols were developed to provide this [Com91], but eventually *IP*, or *Internet Protocol*, was adopted and the Internet was effectively born. Along with IP were adopted several other protocols which used IP to do their low-level routing and the result was the *TCP/IP protocol suite*.

The TCP/IP protocol suite was developed to provide Internet-wide connectivity independent of the physical networks being traversed. TCP/IP was initially proposed in 1974 by Vint Cerf [CK74]. In this initial proposal, TCP and IP were not separate protocols, but the capabilities were combined into a protocol commonly called TCP. Later revisions on the ideas presented by Cerf eventually resulted in the development and full specification of two separate protocols, *TCP* (*Transmission Control Protocol* [JP81b]) and *IP* (*Internet Protocol* [JP81a]). Services supporting process-to-process communication, sequencing, and reliability were moved into TCP, and the addressing structure and general routing issues were moved into IP. It is important to note that IP was adopted for the Internet, but was not designed specifically for the Internet. This is evidenced by the relatively small address space (32 bits) [JP81a] which is quickly being depleted.

The depletion of the address space has prompted interest in the development of a new protocol for the Internet. The current version of IP is known as version 4 (IPv4). Development began early this decade on a new protocol called *IPv6* or *IPng* (next generation). In many ways this protocol is similar to IPv4 except that it provides a larger address space and better mechanisms for handling today's needs [BTB95].

Currently, there are two main types of research being performed on Internet protocols. The first type of research is centered around the IPv4 protocols and their extensions. Extensions such as Mobile IP (which has been designed to support the use of portable or mobile computers) are very new and need to be tested, examined, and refined so that their functionality can be added to the Internet. In addition, the performance of IPv4 and its family of protocols needs to be studied using the variety of technologies currently present in the Internet. Originally the Internet protocols were developed for research, and they have been constantly researched ever since.

The need for research into IPv4 is at least as important as ever because of the development of the IPng protocols. The refinements being introduced into IPv6 are based on extensions to IPv4, and therefore these extensions need to be researched and studied adequately to prevent additional errors being introduced into IPv6. Through thorough investigation of IPv4, the new IPv6 protocols can be made more robust.

The second type of research is centered around developing and testing the new Internet Protocol IPng (IPv6). Although IPv6 is in its infancy, the specifications are quickly becoming draft standards, and implementations are being produced for evaluation. Already successful interoperability tests have been performed between different IPv6 implementations. The further development of IP affects the protocols which rely upon it. Protocols such as TCP need to be redone to support the new environment created by IPv6. As well, new protocols are being developed to exploit and support IPv6 features.

This paper will examine the different aspects of Internet research currently being performed, and will focus on the proposals developed for mobility support by various groups. In particular, the proposal from the *Internet Engineering Task Force*, commonly known as *IETF Mobile IP*, will be studied. In addition the development of IPv6 will be explored and the adaption of IPv4 Mobile IP proposals for IPv6 will be discussed. The sections that follow will further explain terminology related to the Internet protocol suite and networking, and will introduce Mobile IP. The history of mobile IP and the evolution of competing protocols will be presented, and the IETF proposal will be examined. The mobility support section will conclude with a presentation of some current research projects being pursued around the world. Further sections will include a discussion of the IPv6 protocol and this will provide background for a presentation of the current IPv6 Mobile IP proposal from the IETF. This paper will conclude with opinions on the state of mobile and advanced IP research, and a discussion of future research objectives.

3 An Introduction to TCP/IP and Internetworking

Internetworking is the term used for services that provide support for transmitting information across a heterogeneous group of networks. These networks usually implement a wide range of local technologies for direct interaction between computers within each network. An internetwork protocol must be able to provide service in a manner which makes the hardware differences across networks transparent, and in many cases the existence of different networks themselves transparent to the user and many user applications.

This section will provide a brief introduction to the concept of internetworking. Two major internetworking standards exist today, the *OSI* model [DZ83], and the TCP/IP model. This paper will focus on the TCP/IP model for internetworking as it is predominantly implemented in the Internet, and it is the primary target of research efforts into mobility support and other efforts in the Internet community. The discussion of TCP/IP will be presented as it maps onto the OSI model.

The OSI model defines a set of seven protocol layers which define network connectivity. Each layer expects certain services from the layer below and is expected to provide certain services to the layer above. The layers in the OSI model are presented in Table 1. The IP protocol itself lies within the network layer as it is used to support internetworking.

Before internetworking is discussed, the variety of local use networking technologies will be

Table 1: The OSI protocol layers. [Tan88]

Layer	Description
Application	User level program interface support. This layer provides for translation between programs on different machines so that a consistent application environment can be achieved across network connections. Terminal emulation and operating system command translation are two tasks performed by protocols at this layer.
Presentation	This layer is concerned with translating data for different architectures. Different bit combinations may be used for representing the same data element on different platforms. It is the job of the presentation layer to translate data into a standard form when transmitting, and from the standard form into a platform-specific form when receiving.
Session	Protocols providing a connected state between two processes on different machines. Examples of Session layer protocols are those used for TELNET and FTP.
Transport	Protocols which provide connectivity between two processes on different machines. TCP and UDP are examples of these protocols. The transport layer essentially accepts data from the session layer, divides it up into packets, specifies the destination process and machine, and gives the packets to the network layer for transmission.
Network	Protocols which provide host to host connectivity across an internetwork. IP is an example of a protocol at this layer.
Data Link	The Data link layer is divided into two sublayers (Logical Link Control and Medium Access Control). Protocols for the link layer provide connectivity between computers linked by a physical wire (or a bridged connection), and guarantee delivery within a single network over a distance no greater than one hop (i.e., no intermediate hosts involved). The medium access control sublayer (MAC) is used on broadcast networks to manage access to the communication medium or physical layer. The MAC sub-layer makes the physical layer appear dedicated to the host needing access. The logical link control sublayer is closely linked with the network layer and handles flow control, error control, framing, and synchronization. The logical link sublayer makes the underlying physical layer appear error free for internetwork transmission.
Physical	This layer is the actual physical wire and network cards used to implement a physical network. The main concern is ensuring that a 1 bit transmitted is received as a 1 bit and the same for a 0 bit. Concerns revolve around voltages and other low-level aspects of hardware.

hinted at by a discussion of some of the types of networks present in the Internet. Next, the principle agents (types of computers) which make internetworking possible will be presented and discussed. Finally, the Internet protocol itself will be discussed.

3.1 The Underlying Technologies (Local Area Networking)

The Internet is a virtual network that is actually the combination of many smaller networks. A recent estimate of Internet size rates the number of networks at more than 80,000 (as of January, 1996), and these networks include well over 5 million computers generating greater than 16 trillion octets per month of traffic [NSF95]. The sheer size of these numbers should hint at the need for application support which hides differences across network configurations. The communication methods inside each network are independent of all other networks, and a variety of methods have been developed to support local area communication. In this section we will introduce local area networking protocols and systems as a basis upon which internetworking can be built. The networks to be introduced include *token ring*, *wired ethernet*, *wireless*, and *ATM* which are present in the Internet today [Com91].

An individual network is essentially a collection of computers that are linked in some way such that one computer can directly talk to another computer in the group. Such transmissions between computers are usually over a shared medium (such as a radio frequency, or a physical wire) which can support only one message at a time. The inability of the medium to support concurrent communication results in a need for cooperation between different computers in the network. Cooperation is achieved through the definition of rules which every computer must use when communicating with others. These rules together form a *protocol* which uniquely defines how computers access a specific type of network. In the OSI model, protocols responsible for negotiating access to a shared medium are situated at the *medium access control* (MAC) sublayer of the *data link layer*. The data link layer is responsible for producing packets for the network and delivering them to addressable hosts on the network. The link layer (composed of the MAC and LLC sublayers) is fundamentally meant to deliver packets between two hosts linked by a wire.

Various link layer protocols have been defined over the years, both as a basis for comparison and to address certain requirements and capabilities typical of a specific type of hardware. Many of these protocols are synonymous with the hardware they are implemented for. Some well-known network types (and their protocols) are Ethernet, wireless, ATM (high-speed optical networks), and Token Ring. Ethernets are groups of computers that are linked to a common wire which acts like a computer bus. Every computer can talk on the wire and every computer can listen to the wire. The Ethernet CSMA/CD protocols control concurrent access to the wire. Token ring networks are groups of computers that are linked together in sequence in a loop. Control of the network is handled through the use of a token that circulates around the ring. If a host has the token it can transmit, otherwise it can't [Com91]. A wireless network or subnet is composed of a number of hosts transmitting on a common frequency. Sometimes one of these hosts is also attached to a wired network and can act as a gateway between these two networks. The wireless protocols operate much like those for shared medium networks like Ethernet, except that they may also contain mechanisms for handling signal strength fluctuations and other considerations particular to the wireless domain.

ATM is a network protocol for high-speed optical networks. ATM stands for *Asynchronous Transfer Mode*. With ATM, the hosts transmit information in tiny bundles called cells. Many features are built into the protocol, such as performance negotiation, dedicated routes or *permanent virtual channels* (*PVCs*), and different classes of traffic. In recent years TCP/IP has been used effectively over ATM networks. New facilities for traffic classes and performance guarantees in IPv6 will help utilize the facilities offered by ATM [Par94].

In general, link-layer protocols are limited to communicating within a single network (a single group of connected machines reachable within one hop). In order to cross the network boundary, higher level protocols are needed. However, due to the functionality provided by these protocols, higher-level protocols do not need to worry about collisions or hardware specific issues that may affect an individual network.

Higher-layer protocols, such as IP which handles internetwork routing, merely have to tell the link-layer protocol what computer on the local network to deliver the packet to, and the link layer will guarantee that it's delivered. The job of the IP protocols is to route between networks and provide mechanisms so that special machines, called gateways, can communicate. Gateways communicate across individual networks in order to forward packets across the network onto another until they reach their destination. For more information, see [Com91], [Tan88], and [JP81a].

Making an internet work is the job of various entities that comprise the internetwork itself. One of these entities (a gateway) has already been introduced. The following subsections will discuss some of the principle agents which make connecting these diverse networks possible, and will introduce the Internet Protocol handling internetwork communication itself.

3.2 Principle Agents and Important Terminology

The Internet is composed of many entities that make it function. These include the hosts that we use and communicate with, as well as routers, gateways, and other entities. Without an understanding of what some of these entities are, a discussion of internetworking is limited. This section will briefly introduce some of these entities and present some frequently used terminology.

Packets are the collections of data that are transmitted on networks. Packets usually have a header corresponding to a protocol which specifies where the packet is to go, along with other information. The *packet header* is a sequence of bits placed at the beginning of the packet to carry protocol information. The data being transmitted is called the *payload*, or simply the data.

Internet *hosts* are machines with interfaces that in some fashion communicate with other machines on the Internet. Hosts can be computer terminals, personal data assistants (PDAs), or even microcomputers embedded in ordinary machinery. Hosts have a unique identifier or *address* which provides a method for two hosts to find and exchange information with each other on the Internet. Each host has its own unique identifier which is independent of any hardware it has, but which carries with it information needed for finding the computer's location.

Internet hosts are also *routers*, and they can be routers exclusively (i.e., routing is their sole responsibility and they do not allow user login sessions). A router is a machine that keeps traffic flowing on a network or internetwork by forwarding packets to other routers. The basic task of a router is to decide where to send a piece of data it receives for another host and then send it there. In a sense, any host makes a routing decision when it assigns an address to a packet it transmits [Com91].

A *gateway* is a special type of router. Gateways are the routers of the Internet and are used to direct packets across different networks to their final destination. A gateway is a computer with an interface on more than one network or subnet. When a host on a network wishes to communicate with a host on an external network, it must send its packets through a gateway. On most networks there are only a few gateways. One of these gateways is determined to be the default gateway. When a packet needs to be delivered but the address does not map onto a specific gateway, then the default is used. A gateway will accept packets on one interface and transmit them on the other. In this way, packets cross a network boundary. This seems simple, but is actually complex because of the variety of networks on the Internet. While the one interface may be on an Ethernet, the other may be on a token ring or wireless connection. Differences in the size of packets on these networks and in the protocols that are used cause the gateways overhead, which uses up CPU time and causes delay in the packet forwarding.

Each gateway maintains a mapping of Internet addresses to other gateways on the networks it is connected to. When a gateway receives a packet, it examines the destination address and forwards it to the appropriate gateway on the appropriate network. Note that the gateway can forward only onto the networks to which it is connected, so its tables will only record the addresses of gateways on those networks. The various protocols used will not be discussed here, but gateways communicate information to each other by propagating it to neighbouring gateways on networks to which they have have an interface. This limited scope of information exchange is all that is needed for correct routing information to propagate the internetwork. In this way a packet will travel, through the Internet, hop by hop, until it is finally broadcast onto the destination network [Com91].

Protocol layering is a technique for dividing up the task of providing connectivity into distinct layers each of which has a responsibility to the layer above and expects certain services from the layer below [Tan88]. Protocols operating on individual physical connections between machines are called link-layer protocols. IP is classified as a network-layer protocol because the network layer is responsible for internetworking. The layering of protocols can get complicated and somewhat contrived (especially when using the OSI model). In this paper we will consider IP to operate at the network layer, and its associated protocols UDP and TCP to operate at the transport layer. The next section will briefly discuss the TCP/IP protocol suite itself.

3.3 The Internet Protocol Suite

The Internet Protocol Suite is a collection of protocols which are commonly bundled together as TCP/IP. These protocols include IP, UDP, TCP, TELNET, FTP, and others. This section will briefly introduce IP, UDP, TCP and ICMP.

Each of these protocols has a function to perform in internetwork routing. IP is the basic underlying protocol for internetwork routing. TCP is a transport-layer protocol providing a stable connection between two internetwork hosts. UDP is a transport-layer protocol providing unreliable connection over the internetwork. ICMP is an error message transmission protocol for propagating network state information across the internetwork.

3.3.1 IP: The Internet Protocol

IP is the network layer protocol at the heart of Internet addressing. At the IP layer the packet is merely forwarded along the internetwork until it reaches its final destination or is somehow lost. There are usually hardware mechanisms to handle retransmissions in case two packets collide, so packets are not usually lost this way, however they can be scrambled by noisy transmission media and they may be dropped if the network is used beyond its capacity (i.e., there is *congestion*). The IP protocol will accept a packet with a valid Internet address and will forward it through the Internet. There are no guarantees that the packet will arrive at the destination (i.e., IP is an *unreliable* protocol). The source is never warned if a packet has to be dropped, but if a network is unreachable (e.g., a gateway is down) or a host cannot be found, the source is informed of the error using a different protocol called *ICMP* [Com91].

IP accepts packets from higher levels which inform IP as to which host on the Internet the packet must be delivered (i.e., the destination host). The IP layer uses the destination address, along with the address of the sender, an identifier representing the higher-level protocol using IP, and other information to produce a sequence of bytes called an *IP header*. This header is added to the front of the packet and this packet is handed to the lower-level protocols for delivery on the network. If the packet does not have to cross network boundaries, then the IP layer serves no purpose other than to identify the source and destination by their commonly-known permanent IP addresses.

IP originally was designed as an early research project in internetworking. The main aspects of IP were designed for a small internetwork primarily concerned with research. Therefore, IP has various clumsy aspects which detract from its usability. In particular, the desire of the protocol designers to experiment with the Internet caused the header to be defined so that it includes many fields not needed for Internet routing in the vast majority of cases. As a result, certain eccentric routing options are also allowed which add to the complexity of the IP packet. In order to allow these options to be processed, an additional overhead is required. As a result, some implementations of the protocols chose to discard all packets with these options, and other implementations do not correctly implement all options [Myl92].

IPv4 protocols specify individual hosts on the network by a 32-bit number. This 32-bit number, or address, is subdivided into a network address and a host address. There are three classes of networks which are defined in TCP/IP: class A, class B, and class C. Class A addresses have the first octet (set of 8 bits) specify the network, and the last three, the host. Class B networks have two octets for each of the network and host addresses, and class C addresses have three octets for the network address and one octet for the host address. This scheme effectively addresses the large number of small networks (class C) and small number of large networks (class A) found in the Internet [JP81a].

The addressing scheme used in IP has proven to be very effective in reducing the work required to route packets. Extensions to the addressing scheme have resulted in *broadcast* capability (the ability to address all hosts on a network) and *multicast* capability (the ability to address all hosts in a specific group on a network) being supported on the Internet. Unfortunately, the size of the address space is proving to be too restrictive for effective subnetting and global unique addressability in the modern Internet which has grown far beyond what the protocol was intended for.

Despite the limitations of the Internet protocols in IPv4, they have served the Internet well, and through extensions have been able to cope with Internet problems such as congestion, routing loops, and changing global topology. Even the gradual depletion of the IPv4 address space has been slowed by a new approach to assigning network addresses which assigns consecutive class C addresses to large corporations rather than a single class B address (which can consume as many

IPv4 host addresses as 256 class C networks) [FLYV92]. The use of subnetworking to further divide the class B network addresses that are issued [MP85] has also helped to extend IPv4's effective lifespan. However, it is inevitable that the IPv4 address space will be depleted soon, and this is the driving force behind development of a new Internet protocol.

In addition to the increase in the demands on the Internet resources, there has also been a change in the technology that is being used on the Internet. This has resulted in a continual push for research using the current IP version so that new applications can be supported quickly. The Internet today is as always a heterogeneous mixture of many different types of networks. Today, however, there are increased capabilities available that are not exploited by the current Internet Protocol. As well, there are requirements that can not be met by the current Internet without further extensions. Thus, since IP is the basis for internetworking support, it is the protocol that must be changed first when new capabilities arise that require internetworking. Since the current IP version is proving to be less adequate than needed, an improved version is being developed called IPv6 (see Section 10 for details).

3.3.2 ICMP: The Internet Control Message Protocol

The ICMP protocol is used to deliver error messages through the Internet. Essentially, ICMP operates on top of IP because it uses IP routing and packet headers for communication. ICMP packets are special, however, because they themselves do not cause ICMP messages to be generated, and they are of a higher priority than normal IP packets. [Com91]

ICMP messages are generated by routers and hosts on the Internet when errors are encountered while forwarding packets. Typical errors include *DESTINATION UNREACHABLE*, *TIME EXCEEDED*, *PARAMETER PROBLEM*, and *SOURCE QUENCH*. Destination unreachable messages are produced when a gateway encounters a packet with an address it cannot forward to, or when a gateway is required to fragment a packet to transmit it over a network but the *don't fragment* option is flagged. Time exceeded messages are produced when a packet is dropped due to its *time to live* field reaching zero (this is designed to prevent routing loops). The *time to live* field is in the packet header, and is decreased at every router. Parameter problem messages indicate invalid data being passed as arguments in a packet's header. This error indicates a bug in the source or some intermediate gateway's routing software. Source quench messages can be issued by routers experiencing congestion which require senders to throttle their transmission rate so that the router can recover and catch up [Tan88].

3.4 UDP: User Datagram Protocol

TCP and UDP are higher level protocols that operate (sit) on top of IP. These protocols are said to operate at the transport layer and are the interfaces that user programs usually operate with.

UDP is a protocol that provides user-level programs with the ability to communicate across the internetwork. This is different from host-to-host communication in that the identity of programs communicating need to be maintained. Without maintaining the program identities, the destination host of an outgoing message has no way of knowing which program to deliver the message to. Many different program conversations can cause packets to be routed between two hosts at the IP layer where these conversations are reduced to indistinguishable packets.

UDP packets are called datagrams, and essentially UDP is just a higher-level interface to IP.

UDP provides users with access *ports* that are interfaces to individual processes running on a host. These ports are the mechanisms which allow individual programs to communicate with one another over the Internet. Many programs can communicate simultaneously between two computers using different or even the same ports and the messages do not get confused by UDP [Com91].

3.4.1 TCP: Transmission Control Protocol

TCP provides user programs with port access, but also provides reliability assurance. When a TCP connection is set up, certain communication parameters are negotiated regarding how many packets can be sent in a row, and other things. TCP provides mechanisms for this negotiation, but also provides an acknowledgement scheme in which every packet sent from host A to host B has to result in an acknowledgement being sent from host B to host A, and vice versa. If acknowledgements are not received after a certain period of time, the packet is assumed to be lost, and it is retransmitted. The mechanisms for TCP were designed and have been refined to work well in the standard Internet. These mechanisms do not always work efficiently with new technology and are the source of much research today [HN87].

4 Concepts in Computer Mobility

Mobility is a user behaviour made possible by the increasing availability of smaller portable machines and ease of connection to the Internet. Mobile computing is a research field that has grown to address the issues surrounding user mobility. Supporting mobile computing presents many challenges, including the management of physical location changes and maintaining user connectivity regardless of where or how he/she is connected.

Mobile computing is about managing mobility issues with respect to an internetwork. The underlying link-layer support can include wired ethernet, cellular phone technology, wireless transmitters, or PPP/SLIP dial-in sessions using a modem. Each of these different techniques for connecting to the internetwork have specific problems (some of which were discussed earlier). Addressing the problems relating to the access technology should be handled at the link layer unless network boundaries are crossed. Since much research effort is concentrated on computers with wireless interfaces, these computers are introduced in subsection 4.1.

In general, mobile computing devices can be categorized in three ways: desktop computers, portable computers, and embedded computers [Chi95]. Portable computers include many types of computing devices now available on the market. These devices range from fully functional laptops to personal digital assistants (PDAs). Laptops can perform most of the functions supported by desktop computers. As a result, users used to easy network connectivity with their desktop workstations desire this with their laptops as well. PDAs are more limited in functionality, and are used to store personal information and exchange this information with a stationary computer. Embedded computing devices are even more limited in power.

Embedded computers are computational devices which are part of many different machines today, including cars, trucks, military vehicles, and civilian airplanes. It is theorized that with the increased address space provided by IPv6 (see Section 10.2) many of these machines may acquire Internet addresses to facilitate their communication with support entities for the purpose of transferring geographical, tactical, status, and other types of information [BTB95].

All portable machines are becoming increasingly powerful in terms of both processing power and long-term storage. The fully-functional desktop stations which change location are truly as powerful as their fixed counterparts and do not need special consideration for their lack of processing capability.

Due to the fact that the Internet supports not only computers as we know them, but computation devices capable of sending and receiving messages, we will use the term *host* to describe an entity with an interface to the Internet. A mobile host is a machine with an Internet connection that is not always connected to the Internet by the network corresponding to its IP address (its *home network*). A mobile host is distinguished by its ability to cross network and subnetwork boundaries while still maintaining its connectivity in a fashion that is semi-transparent to layers above the network layer. A mobile host can be connected to the Internet in the same way as any fixed host, using ethernet, fiberoptics, dial-up lines, or even wireless interfaces. However, the presence of a wireless interface does not by itself make a host mobile in the sense discussed here.

4.1 Wireless Interface Computers

Wireless Interface computers are commonly called wireless computers, and are frequently referred to as mobile computers. Much of the mobile computing field is connected to addressing concerns associated with wireless interfaces. These computers are truly mobile computers. They do vary location across time, but to say that a mobile computer is a computer with a wireless interface is false. To show this we will examine a wireless interface computer.

Wireless computers are characterized by their ability to maintain network connectivity while they are physically moved from one location to another. The key concept is that motion and connectivity exist at the same time. This is permitted by the wireless communication medium which allows the wireless computer to keep in contact anywhere within the range of its transmitting base station. Once a wireless computer is beyond the reach of its base station it must either find another base station, or lose connectivity. The transfer of control from one base station to another is called hand-off, and is handled at the link-level. Some Mobile-IP proposals have a basis in this hand-off, as sometimes crossing a wireless boundary also means crossing a network boundary.

Hand-offs from one wireless transmitter to another do not necessarily indicate changes in the network to which the wireless computer is connected. For example, suppose that a business has a local area network (LAN) connecting its offices. Connected to the local network at various points are wireless base stations. Such a LAN in many instances will be a single network or subnet. This means that all computers on that network have the same network address, and therefore all Internet routing is the same for the computers. The only routing differences arise at the link-level. In such an environment, the wireless base stations act as bridges for the LAN, and not as gateways for an internetwork. As a result, wireless computers can roam freely in the LAN without creating great difficulty at the network layer and without causing a loss of connectivity with the Internet. If instead of one large connected network, the business LAN is composed of many subnets, then hand-offs can cause problems with network routing because each subnetwork will have a different routing address for packets. More details of the routing support problems are given in the next section.

5 Providing Support for Mobile Computing

The Internet is no longer the experimental tool used exclusively for research, scientific investigation, and military communication that it once was. It has now become a tool widely used by many professionals and businesses. In particular, e-mail has quickly become the most economical and effective way to communicate with many people across large distances in little time. These benefits are being seen by the majority of professionals, and there is a growing number of highly mobile people who use the Internet and rely on its continuous availability for vital services (such as e-mail) [FZ94].

Today's highly mobile users (this certainly does not preclude either the researcher or the military user) desire the ability to take their networking capabilities with them unchanged as they relocate. General computing capabilities are facilitated by the use of portable computer systems. These systems are smaller, more limited versions of their desktop counterparts which can be used in any location (much like a powerful calculator). These computers work well for local tasks, however once Internet access is needed, the solutions become either expensive or cumbersome. Essentially, the mobile user wanting Internet access has four choices, only one of which offers direct Internet connection:

1. The user can subscribe to an Internet provider near his/her home location. When the user is away from his/her Internet service provider he/she can dial into the service provider using a long-distance telephone service to achieve connectivity.

2. The user can purchase a subscription to a remote Internet provider at various locations to save on the long distance charges.

3. The user can apply for new Internet addresses for every remote location he/she visits and can use these addresses (once connection to a network is established) to communicate over the Internet.

4. The user can connect to a free Internet service which usually has limited capabilities and requires an application for a new account.

Although all of these options are available (and are used [Rei94]), none of them provides truly transparent migration on the Internet. Since all require that the mobile host change its address with every move. The implications of address changing include the inability to continue conversations across connection moves, and the inability of remote hosts to remember where the mobile host is if it provides some service that is needed.

The problems with mobility support arise because the addresses used in the Internet contain information about the physical location of a host. This information is carried in the network portion of a mobile host's permanent home address. This network address tells other hosts in the Internet in which direction they should forward packets in order to have them arrive at the network containing the destination host [Com91].

At the link layer, mobility is already supported due to the lack of location information associated with an individual host on a single network. In cellular networks, a protocol called *Cellular Digital Packet Data* (*CDPD*) has been developed to manage mobile hosts exchanging computer data rather than voice. This type of network is different because of its size, and in fact different base stations may have different Internet network addresses associated with them. Because of this, the CDPD protocol has been defined with mobile internetworking in mind, and it has many of the facilities provided by Mobile IP extensions. This protocol is limited to cellular networks, however, and it is built to handle the specific problems associated with cellular network technology, but doesn't support wired or other technology. Since CDPD has been developed by private companies, it has emerged quickly onto the marketplace and is a viable option for users wanting mobile e-mail access and other simple data exchange capability. Due to its focus on cellular networks, and its lack of focus on network-specific routing, CDPD is not the general solution that is needed for the Internet.

A comprehensive solution to the location problems of mobile host connectivity starts with providing network layer support for physical location changes. This support must make the location change as transparent as possible to higher layers which rely on the mobile host having a constant address. Thus, although the host may be temporarily unavailable to the Internet, when it is reachable, it can always be reached using the same address. The following sections will focus on Mobile IP proposals which have been designed to support static addressing of mobile hosts. These sections will present a general discussion of mechanisms used in mobile IP, the terms used for describing mobility, and the desirability of mobile IP.

5.1 The Need for Mobile IP

Basic mobility is easily achieved within a single network or subnet. Mobility within a network is independent of Internet protocols and their problems with location-dependent addresses. Disconnecting a computer and moving it to a new physical location is fine, as long as no network boundaries are crossed. In this situation, connectivity can be managed by the link layer protocols. As mentioned earlier, hosts within a single network or subnet are usually linked together via a common medium and communicate using hardware addresses and link-layer protocols. At this level, traffic is usually received by all hosts on the network and is filtered by the hardware interface itself. Since all traffic is received by all hosts, the physical location of a host in this environment is unimportant.

When a mobile host crosses a network barrier, however, packets directed to it can no longer reach the host via standard Internet routing techniques. When packets are directed to the host, they are routed using the network part of the host's address, and therefore they are sent to the wrong network. There are two solutions to this problem: remove the location dependency from the Internet addressing scheme or build in support for forwarding messages to temporary locations. Due to the success of the Internet protocols, and their large installed base, it is not feasible to change the Internet addressing scheme in such a radical fashion (also, it has been an efficient way of encoding network location, making the job of routing easier). Therefore, the need is for a forwarding mechanism.

In the 1990s, much research effort has been applied to the specification of extensions to the TCP/IP protocol suite to support mobility across networks. Some of these approaches have advocated changes to the routers which exist in the Internet today, and rely on these changes for their full functionality. Other approaches, such as IETF Mobile IP, are based on having a stable home base or network which can forward information to a recognized foreign network where the host is residing. These last approaches can exist in an Internet where routers know nothing about mobility, but changes to the routers themselves can theoretically enhance performance.

The current number of hosts and networks in the Internet is very large. Most of these hosts provide facilities for routing packets, and many hosts function as routers in helping a packet reach its destination. If supporting mobility required changing all these hosts, it would be impractical, and not implemented as a result. As well, the routing mechanism present throughout the Internet works quite well, and a solution to the mobility problem is needed which exploits the facilities for routing already present in the Internet while not requiring immediate change to the routing computers involved.

One can see that there is a need for mobility support in the Internet. Network connectivity is commonly accessible in desktop computers, and is desirable for the mobile user as well. As the use of portable computing devices increases, so will the demand for cheap support for easy connectivity with little or no disruption of service as the computing device migrates on the Internet. Since IP is the predominant protocol used for communicating on the Internet, there is a need for an extension to IP which will support the functionality desired.

In order to study the approaches that have been taken to provide mobility support, certain terminology needs to be defined. This next section will present some of the terminology used to describe mobile IP proposals. Attempts have been made to try and make the discussion generic to all protocols, but where different terms exist for the same entity, IETF Mobile IP terminology is generally used.

5.2 Mobile IP Terminology

This subsection will present some of the terminology used when talking about Mobile IP protocols. Some protocols expand upon this terminology, and others are more restrictive. The following is a basic set of terms:

Mobile Host (MH) A host that will vary its location by crossing network boundaries while still requiring full connectivity through its permanent Internet address. These hosts must be aware that they are mobile and must be able to interact with mobile routing agents (see below) so that these agents can set up special packet routing needed to deliver the MH's packets to the its temporary locations on foreign networks (see below).

Home Network (HN) The network whose network prefix corresponds to the mobile host's home address (i.e., the mobile host's home address has been assigned on this network). [Per96]

Foreign Network (FN) A network other than a mobile host's home network that the mobile host uses for connection to the Internet. [Per96]

Home Address A mobile host's permanent address on its home network. This is a unique identifier used to correspond with a mobile host.[Per96]

Care-of Address An address that a mobile host acquires from a foreign network to use as a forwarding address while visiting that network. Packets to the mobile host are encapsulated by a mobile routing agent and sent to this address. Facilities are provided at the care-of address for delivering the packets to the mobile host.

Mobility Binding An association of a care-of address with a home address. [Per96]

Mobile Routing Agent (MR) A host on the Internet which can intercept packets on behalf of the mobile host and ensure their delivery to the mobile host on its foreign network. These routing agents are aware of both the permanent and care-of addresses associated with a mobile host and implement the routing between these two addresses. Some Mobile IP protocols have only two mobile routing agents (a home agent and a foreign agent), and others have entire networks of modified routers.

Correspondent Host (CH) Correspondent hosts (CHs) are computers that wish to communicate with the mobile host. These computers know the permanent address for the mobile host, and sometimes can be made aware of care-of addresses that the mobile host may be using for communication. Packets from the mobile host to a correspondent host are routed normally, but packets from the correspondent host to the mobile host must be intercepted and redirected by MRs.

Tunnel The path followed by a packet while it is encapsulated. Tunnels are commonly thought of in connection with the encapsulation that happens at both ends of the tunnel. A mobility agent will encapsulate a packet to the mobile host by enclosing the entire packet with an outer IP header that specifies the care-of address at the other end of the tunnel. This tunnelled packet is delivered using normal IP mechanisms, and is decapsulated at the other end of the tunnel. [Per96]

Visitor List A list of visiting mobile hosts maintained by a routing agent on foreign networks. This list contains information such as the home agent for the mobile host, a time-out value for deleting this entry from the list, and possibly the network interface to which the host is connected, and its network address.

Mobile Security Association (MSA) A relationship set up between two mobile entities which promotes correct entity identification. This helps to ensure that a host is the mobile entity that it says it is.[Per96]

This subsection has introduced some of the basic Mobile IP terminology. The next section will introduce IP-in-IP encapsulation which is the primary form of encapsulation used for redirecting packets to a mobile host.

5.3 IP Forwarding of IP

The mechanism which makes Mobile IP possible is called *packet forwarding*. Packet forwarding involves taking a packet addressed to a host's permanent location and forwarding it to a different location. This means a packet sent to the mobile host's home address can be intercepted by another computer and redirected to the current location of the host. In this interception process the original packet must be preserved because higher-level protocols, and IP itself, depend on information in that packet for correct processing at the destination.

At first, it may appear that forwarding would be easy: simply change the destination address for the packet to the temporary one, and send it away. There are many problems with this approach. First, most of the time the temporary address associated with an MH will be another computer that will forward the packets onto a local network where the MH resides. Non-encapsulated packets will be interpreted by this host as packets destined for itself, and will not

IP Header	IP Data	$-->$	Hardware Header	IP Header	IP Data

Figure 1: IP encapsulation within hardware packets.

IP Header	IP Data	$-->$	IP Header	IP Header	IP Data

Figure 2: IP encapsulation within IP packets.

reach the mobile host. As well, even if the packets do reach a mobile host (i.e., if the care-of address is temporarily assigned to the MH itself), there is no guarantee that the temporary address for that host will still be valid at the time of delivery for the next packet (i.e., the host may move again and a new host may be using that address).

The mechanism for forwarding commonly used on the Internet is encapsulation. Encapsulation means taking the entire data packet and placing it inside a new data packet (see Figure 1) by adding a new header onto the front of the packet. This is precisely the mechanism by which IP traverses individual networks. The IP packets are encapsulated in network packets using the link-level protocol and are forwarded over the local network.

For implementing Mobile IP a similar method of encapsulation is needed. The difference in mobile computing is the need to encapsulate IP packets within IP packets (IP-in-IP). The IETF has produced Internet standards for IP-in-IP encapsulation ([Per95a] and [Per95b]). This type of encapsulation makes it easy for the packet to be redirected fully intact to the mobile host. A disadvantage of using IP-in-IP encapsulation is the overhead that is involved.

The overhead with encapsulation is of two forms. First, there is processing time needed for encapsulating and decapsulating. Secondly, there is the space requirement needed for the encapsulating header. The addition of an extra IP header into the packet increases the size of the packet itself. Because of hardware and other concerns, individual networks define a value called the *maximum transmission unit* (MTU) which is the largest packet size that network will accept. The increased packet size of the encapsulation packet can cause the MTU of the transmission medium to be exceeded. This causes *fragmentation* of the packet, and the overhead associated with fragmenting and defragmenting. In addition, because the original packet is encapsulated, ICMP errors messages cannot be properly propagated back to the original source of the packet without special measures being taken [CWB96].

In some Mobile IP implementations, encapsulation is used by special routing agents outside the home network which intercept packets addressed to the mobile host's permanent address and forward them to the network on which the mobile host resides. This forwarding may be one step or many if the mobile host has moved and there is inconsistent routing information in the network. Current research is investigating the optimization of packet routing to a highly mobile host through multiple forwarding and caching [HBK95]. More complete discussions of Mobile IP routing proposals are presented later in this document (see Section 6 and Section 7). IP encapsulation, as it is performed in many Mobile IP proposals, is illustrated in Figure 2.

IP Header	IP Data	$-->$	IP Header	Min-IP Header	IP Data

Figure 3: Minimal Encapsulation of IP Packets

5.4 Minimal Encapsulation

Minimal encapsulation is a refinement of normal IP-in-IP encapsulation designed to reduce the size overhead of packet encapsulation. Size overhead can become a problem especially in lossy environments like wireless. The concern for reducing header overhead results in a highly minimalized header which does not even include space for fragment information. Therefore, minimal encapsulation cannot be used if the packet has already been fragmented.

Minimal encapsulation specifies a method by which the original IP header is modified slightly and reused. A second, minimal forwarding header is added to the packet between the outer header and the packet data. This type of encapsulation is shown in Figure 3. The original protocol number, destination address, and source address are copied into the minimal forwarding header from the original packet header. The new modified outer header contains the original data except the protocol number is changed to 55, the destination address is changed to the care-of address of the MH, and the source address is changed to the address of the encapsulating agent.

5.5 IP Encapsulation of Broadcast and Multicast Packets

The encapsulation of normal IP traffic can be handled quite easily using the IP Encapsulation method outlined in the section above. Broadcast and multicast traffic each present a unique set of challenges for the Mobile IP protocols. This section will discuss proposed methods for handling broadcast and multicast packets.

Broadcast traffic is traffic that is sent to a specific address specified by the local network address with a sequence of ones for the host address. An example of a broadcast address is 128.233.255.255 which is the broadcast address for a network at the University of Saskatchewan. A packet sent to the broadcast address is delivered to and processed by every host on the network. Since the network address is local, these broadcasts do not propagate beyond any gateways to other networks.

By definition, broadcast packets are sent to every host with an address on the broadcast network. This includes mobile hosts. The problem is that broadcast packets are not sent specifically to the MH, so mobile routing agents may not forward the packets. It is simple enough for an agent on the home network to encapsulate and forward the packet to a care-of address. When the packet is decapsulated at the other end, however, the broadcast address does not correspond to any MH and the packet is discarded. In fact, unless the decapsulating agent is provided with the netmask of each MH it supports, there is no way determine if the packet even is a broadcast packet and therefore it must be discarded.

The solution for broadcast packets is to encapsulate them twice. First the packet is encapsulated with a header in which the destination address is the MH and the source address is the original source. The remainder of this header is unimportant except that it must be marked to let the MH know it is encapsulated. The outer header has a source address corresponding to the

IP Broadcast Hdr	IP Data	−− >	IP Hdr	IP Hdr	IP Broadcast Hdr	IP Data

Figure 4: IP encapsulation of broadcast packets.

mobility agent on the home network, and a destination address corresponding to the mobility agent on the foreign network. A diagram illustrating this encapsulation is shown in Figure 4.

Multicast packets are even harder to deal with than broadcast packets. Multicast traffic is delivered to all hosts on the internetwork who have subscribed to the particular multicast group to which the packet has been addressed. The delivery of multicast packets is usually controlled by mechanisms that assume a static location for all hosts expecting delivery. This, of course, presents a problem for mobile hosts.

There are multiple ways to handle mobile multicast traffic, some of them particular to the mechanisms used to implement the multicast routing itself. Three general solutions are: *foreign subscription*, *bi-directional tunnelling*, and *remote multicast extension*.

Foreign subscription is the simplest of the three options. If a MH is going to be stable on a foreign network for an extended time, this may be the best option. The MH is required to re-subscribe to the multicast group from the foreign network. If a co-located address is being used, then this address should be the source of the subscription message. If a co-located address is not used, then the home address must be used. This can cause problems with some multicast routing protocols, however, since some multicast routers are required to drop any packets originating from a path which is different from that used to send packets on (see [CWBM96]). This would mean that a MH connected in such a way can receive multicast datagrams but may have trouble sending them. In addition, if the MH is highly mobile, then packets will be lost due to the large set-up time associated with multicast subscription. This option requires no special encapsulation needs.

A second option is bi-directional tunnelling. With bi-directional tunnelling, the mobile routing agent on the home network must also be a multicast router. Using this option, subscriptions are done through the MR on the home agent. When the MH is away from home a bi-directional tunnel to the home MR is set up. This allows both sending and receiving of multicast datagrams with the same guarantees given to fixed hosts. A disadvantage is that if multiple MHs belonging to the same multicast group then the multicast packets will be duplicated on the foreign networks, thereby negating some of the advantages of using multicast in the first place. The encapsulation support needed for this method is essentially the same as that needed for broadcast datagrams. When forwarding multicast datagrams to the MH, the HA must first encapsulate the multicast datagram in a unicast packet destined for the MH. This is done to ensure that the FA will know who the packet is for as it will likely not recognize the multicast address, or recognize the MH as belonging to that multicast group. Once the packet is unicast encapsulated, then it needs to be encapsulated using IP-in-IP as usual. Packets travelling the reverse route from the MH to the HA are multicast packets encapsulated with a unicast header. It is assumed that the HA is a multicast router and knows how to decapsulate and deliver the packets.

The third option is remote multicast extension. This is an option proposed by Vineet Chikarmane [CWBM96]. This option requires a lot of sophistication on behalf of mobile routers on the foreign network (hereafter called foreign agents). Packets from an MH are handled via a bi-directional tunnel as in option two. However, multicast packets destined for MHs are delivered to the multicast address itself on the foreign network. This requires the foreign agent to accept encapsulated packets to a multicast address. In addition, when multiple home routing agents are forwarding packets for a single multicast group, the foreign must be able to designate one of them to be the sole router for the multicast group. A disadvantage of this scheme is that it increases dependency upon a single home router and thereby decreases the robustness of the system. On the foreign networks, mobile hosts accept the multicast datagrams (after decapsulation and link-level broadcast) as they would normally on their home networks. This option is discussed in more detail in Section 7.3.2 and is fully described in [CWBM96].

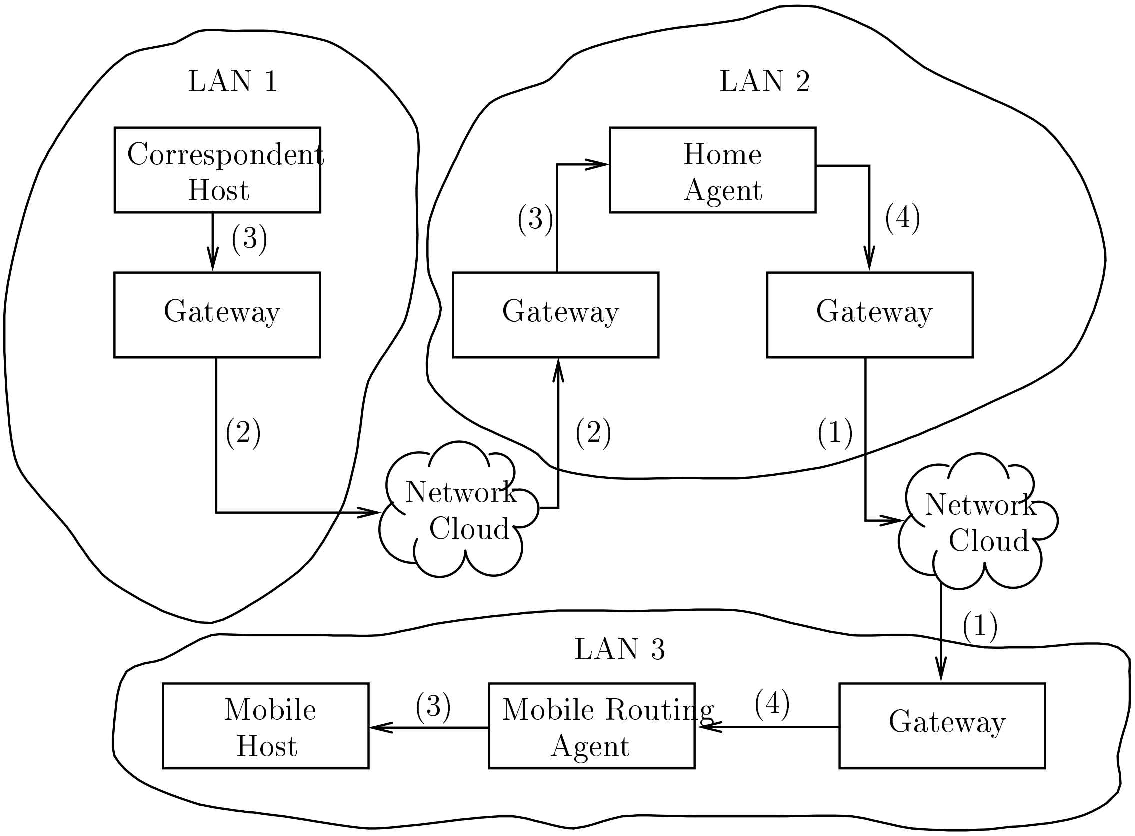
5.6 Simple Dog-leg Routing

In a Mobile IP using dog-leg routing, a packet is sent from a correspondent host to the mobile host's home IP address. The mobile host is known only by its home address, so the packet is delivered to the home network. If the mobile host happens to be there, the packet is delivered normally. If the mobile host is not present on the home network, a mobile routing agent on the home network will accept the packet for the mobile host and will encapsulate it. The destination address in the encapsulating header will be a care-of address for the mobile host, and the source address is usually another routing agent on the foreign network.

Encapsulated packets are routed correctly using normal IP routing. The encapsulated packet is data like any other data. The only differences are at the encapsulation source and the decapsulation destination. The source for an encapsulated packet must produce a new header and a new packet as discussed earlier. At the destination, the host must be able to detect that this packet contains encapsulated information, and should remove the outer packet header and forward the interior packet to the destination host. In performing the decapsulation and forwarding, the host is acting like a gateway to the network on which the mobile host is located. In some cases, such as where the mobile host is connected using a wireless interface, the foreign mobile router may in fact be a gateway to a wireless network or subnet. If this is the case, then the final packet to the MH's hardware interface is only transmitted once on the subnetwork that the MH is attached to.

If the foreign agent handling an MH is not a gateway on that subnet or network, then it will receive the IP-encapsulated IP packets sent from the actual gateway for the network or subnetwork the routing agent is on. The routing agent finds the true destination of the packet using the encapsulated header and retransmits using the hardware address it associates with that host. This means that the packet is transmitted twice on the subnet to which the MH is connected in this situation.

An example of routing to a mobile host is illustrated in Figure 5. In this figure, four types of routing are demonstrated: IP, IP-in-IP, and LAN forwarding of voth IP and IP-in-IP packets. With unenhanced mobile IP routing, a packet originating at a CH is given an IP header with its destination as the MH's home address. This packet is encapsulated in a link-layer packet and forwarded to a local gateway (3). This gateway and others forward the packet using IP routing to the home network of the MH (2). The IP packet is forwarded by link-layer protocols on the home network to the home agent (3). The home agent now encapsulates the IP packeet using another IP header and produces an IP-in-IP encapsulated packet. This is forwarded to an appropriate local gateway (4) that forwards the packet on the interwork using an IP-in-IP tunnel (1). The gateway on the foreign network receives the packet and forwards it to the mobile routing agent (4). This agent decapsulates the IP-in-IP packet and forwards the bare IP packet to the MH using



(1) IP-in-IP (3) Local network routing of IP

(2) IP (4) Local network routing of IP-in-IP

Figure 5: Mobile host routing overview

local link layer protocols (3). A recent proposal has been developed for using a similar mechanism for packets originating from the MH [Mon96]. This is called bi-directional tunnelling, and can be beneficial when faced with firewalls and other security mechanisms that refuse to forward packets whose source addresses do not match the networks from which they originate.

6 The Early Development of Mobile IP

The development of Mobile IP protocols started in the early 1990's, with four major proposals being published in and around 1991. These proposals included Columbia University mobile*IP, Sony Labs VIP, IBM's Protocol using *loose source routing* (also know as LSSR), and Harvard's Protocol. Each of these approaches had its own strengths and weaknesses. The early versions of the first three are contrasted and compared in a paper by Andrew Myles and David Skellern

[MS93b]. The rest of this section will discuss these early protocols. A good source for more information on early proposals is [Myl94].

*Mobile*IP* was developed at Columbia University by John Ioannidis [IDM91], and is based on virtual mobile subnets with a network or internetwork setting. Packets are forwarded to the mobile hosts through the use of IP-in-IP (see Section 5.3).

Virtual IP (*VIP*) was developed at the same time at the Sony Computing Lab by Fumio Teraoka. VIP is based on propagating special routers throughout the internetwork which can cache a mobile host's location and tunnel packets using a new IP option [TYT91].

The IBM proposal was designed by Charles Perkins and Yakov Reckter. This protocol was based on loose source routing which is implemented as an option in IPv4. The advantage of this proposal is that it takes advantage of existing standards for IPv4. Unfortunately, loose source routing is not implemented on all platforms, and is often implemented incorrectly. Perkins and Reckter also proposed a generic architecture that was compatible with all three of the major proposals for mobility support presented above [PR92].

These three proposals and the general architecture were examined in [MS93b] and [MS93a]. After these proposals were investigated, they were found to have various deficiencies. This revelation led to the development by Andrew Myles and Charles Perkins of a new mobile protocol, MIP [Myl93]. While MIP solved many of the problems with the previous protocols but still had some problems including security. While MIP was being developed, David Johnson at Carnegie Mellon University independently developed the Mobile Host Routing Protocol (MHRP) [JMP94]. MHRP used ideas which were similar to loose source routing, but the method of encapsulation did not involve IP options.

Although MIP and MHRP were reasonably robust proposals, they were weak in the area of security. This prompted the development of the Internet Mobile Host Protocol (IMHP) [JMP94] by Andrew Myles, Charles Perkins and David Johnson. This new proposal drew on experienced gained with the other proposals, and incorporated new security mechanisms. One of the new security mechanisms developed was a concept called simple authentication. This protocol has become the basis for the IETF Mobile IP drafts, and its features have been mapped onto IPv6 to provide the basis for mobility support in the next generation of IP protocols [PJ96].

6.1 Columbia University

The Columbia proposal was developed to address mobility on a University Campus. As a result, the proposal suffered from a lack of scalability to the larger Internet domain. There were attempts to address this shortcoming in later versions of this protocol [IDM91], and they were marginally successful. Only the basic protocol and its contributing ideas will be discussed here.

Columbia Mobile*IP supports mobility through the definition of a virtual mobile subnet which is spread across a small number of physical subnets. This idea of using a virtual subnet for tracking mobile hosts was proposed for the Arpanet as early as 1980 [SP80]. Routing in the Columbia virtual subnet is supported by *mobility support stations* (*MSSs*) which correspond to the MRs mentioned in Section 5.2. These MSSs are special routers which act as gateways between the physical subnets and the virtual network.

Mobility across physical subnets is supported by having MSSs in each physical subnet. The mobile host itself is assigned a single unchanging address on the virtual subnet. Other hosts either in the same subnet or outside can always address the mobile host using this address.

The Columbia Mobile*IP specification identifies four cases for interacting with the mobile host. When describing these cases, the terms MHa, MHb, MSSa, and MSSb are used to distinguish between similar entities.

MHa to MHb on same subnet: In this case, communication between two hosts is handled directly by the subnet and its normal address resolution mechanisms (e.g., ARP).

MH to a fixed host: To route these packets, the proposal requires the mobile host to use its MSS as a gateway to the physical subnets. The MSS will use normal IP routing to deliver the packet.

Fixed host to MH: In this case, the packets from the fixed host are routed to any nearby MSSs (which are gateways to the virtual subnet). This MSS (MSSa) then encapsulates the packet and delivers it to the MSS managing packets for the subnet where the MH is located (MSSb). MSSb decapsulates the packets and then delivers them directly to the MH.

MHa to MHb not on same subnet: This case can be thought of as a combination of cases 2 and 3. MHa uses MSSa as a gateway to the network. MSSa forwards the packet to MSSb. MSSb decapsulates the packet and delivers it to MHb.

All of the above assume that the mobile host information has been propagated to the MSS hosts involved in routing the packets to and from the MH. There are two main reasons for this not being the case: the MSS controlling the subnet has never encountered the MH, or the MH has moved onto a foreign network (i.e., intercampus routing). These cases both cause added traffic to be generated on the network because router information has to be exchanged to find the mobile host.

When an MSS receives a packet destined for an MH it does not know, the MSS must contact other MSSs to retrieve the location information for the MH. The process of collecting MSS information regarding mobile hosts is called *Peer MSS Discovery.* The basic idea is that when an MSS encounters a mobile host it does not know, it will broadcast a WHOHAS message to all other MSSs on the network. The home MSS for the mobile host will respond with the appropriate information.

When an MH moves onto a foreign network, the network address will not be the same as that for any MSSs in that network. Therefore, communication with local MSSs will be impossible given the usual methods. Movement can be facilitated if the foreign network supplies transient addresses to mobile hosts located there. This allows the mobile host to communicate with any MSSs and subsequently inform its MSS on the home network of its location. Packets sent to the MH while it is on the foreign network are captured by the home MSS, forwarded to the foreign MSS, and sent directly to the MH.

In order for all this routing in the virtual subnet to work, location information about the mobile host must propagate across all relevant MSSs. Communication between MSSs is handled by the *Mobile Internetworking Control Protocol* (*MICP*). MICP defines the following message types for tracking mobile hosts:

MH Greeting: A message from an MH to an MSS to acknowledge that the MH is on the MSS's subnet and is using that MSS for routing.

Greeting Acknowledge: A message from the MSS to the MH acknowledging the MH's presence.

Forwarding Pointer: A notification to an MSS from an MH informing the MSS of the MH's care-of address.

Forwarding Acknowledge: Acknowledgement from an MSS of a forwarding update sent by the MH.

Redirecting: A message sent from MSS to MSS when a packet arrives at an MSS which no longer hosts the MH but did in the past. This message is meant to update stale location cache entries in the network.

Expired MSS Information: This message is sent by an MSS when an encapsulated packet for a mobile host not serviced or known by the MSS arrives. This message informs the sender that the MSS does not handle packets for the desired MH.

WHOHAS: This message is broadcast from an MSS with no information on an MH but with a packet to deliver to it. This message is broadcast to all other MSSs to determine who is serving the MH.

IHAVE: Information sent by an MSS in response to WHOHAS messages when the MSS is serving the MH. [IDM91]

In a small network, this proposal works fine, but due to the large number of messages needed to track a mobile host's location, this proposal is impractical for the Internet. There have been refinements to the protocol, such as an MH serving as its own MSS on a foreign network and MSSs talking directly to neighbours rather than broadcasting to all MSSs. These refinements do not address the message overhead and the router changes required. The combination of large message traffic, the need for special routers, and the requirement for transient addresses are problems because they put stress on the network capacity, the address space, and also requires the propagation of many routers capable of handling the virtual subnets.

6.2 Sony

The Sony proposal is called virtual IP (VIP) and it is also based upon the idea of a virtual subnetwork existing logically on top of physical subnetworks. Each mobile host is assigned a permanent home address and thereby a permanent home subnetwork in the virtual address space. Hosts are also assigned physical addresses which correspond to physical subnetworks to which they may be attached.

The Sony developers envisioned VIP as an extension to IP which defined a new protocol layer lying above the network layer (including any internetwork layer) and below the transport layer. The virtual network layer hides the physical subnet from the transport layer and provides migration independent services on the virtual subnets. The VIP proposal defines four services which are critical for the transport layer to provide to the session layer, regardless of any host migration [TYT91].

Transport Entity Addressing: The session layer needs a migration-independent identifier for mobile hosts.

Connectionless-Mode Communications: The transport layer must provide datagram service to the session layer, regardless of migration.

Connection Establishment: Transport-level connections must be fully supported, regardless of migration.

Connection-Oriented Communications: The transport layer must be able to maintain sessions in the presence of migration.

In order to address the requirements, the *virtual network layer* was designed to support the following capabilities [TYT91]:

Migration-Transparent Host Migration: A migration-independent address for the MH must be supported.

Migration-Transparent Connectionless-Mode Communication: This service must be supported by virtual-network-layer protocols.

Disconnect Request: Supports preparation for leaving a physical subnetwork when the host is to be disconnected.

Disconnect Indication: Support for notification by the data link layer of impending disconnection.

Connect Request: Preparation for connecting to a subnetwork.

Connect Indication: Supports data-link notification of on-line migration.

This system relies on special routers to intercept the packets destined for an MH and forward them to the MH's current location using its physical network address. Each VIP router has a cache for holding mappings from virtual network addresses into physical network addresses. These caches are updated in three cases. First, when a mobile host changes location, it may broadcast messages to all nearby routers to update their caches. If these routers hold invalid entries, they forward the update to all their neighbouring VIP routers. In this way, updates are propagated throughout the system. These messages are also sent if a router forwards a packet to the wrong physical subnetwork. Cache entries can also time out if the router does not receive update messages within a certain time frame. In all cases, a router will either insert a new entry, invalidate an old entry, or update an existing entry.

VIP requires the MH to acquire a new network address on any foreign network it visits. This is needed because foreign agents which can intercept packets on the MH's behalf are not defined. In order to support this, networks have to keep blocks of addresses free so they can be assigned to mobile hosts which enter the network. In some networks this requirement is infeasible given constraints on the address space.

The packets in VIP are routed via the temporary addresses which are placed in the IP header. The VIP source and destination addresses are placed in an IP option field. Although this forwarding mechanism eliminates the need for IP-in-IP encapsulation, it does require correct implementations of IP options in Internet routers. Unfortunately, many routers do not have options implemented correctly, and some implementations of TCP/IP do not even accept options and drop all packets using them [Myl92].

The need for router propagation is very large in the Sony proposal. This is mostly because the propagation of cache updates depends upon routers understanding the protocol. Without widespread implementation, the cache updates stop very close to the home networks, and routing delays during migration are prohibitive.

Since its early proposal, the Sony system has been further investigated [TUSM94] [TU95]. In fact, the VIP system is one of the proposals being considered for mobility support in IPv6. In the case of IPv6, many of the concerns about VIP disappear because in a pure IPv6 environment, all routers will be capable of handling VIP protocols. The problem, even with an IPv6 implementation, is still backward compatibility since there will be IPv4 routers present in the Internet for a long time to come. See Section 11.2 for more information on VIP development for IPv6.

For the development of mobility support in IPv4, the developers of the system were active in the Mobile IP working group and helped to develop the IETF Mobile IP proposal. This proposal is discussed later.

6.3 IBM

IBM's Mobile IP system was developed by Charles Perkins and Yakov Reckter. They first proposed their ideas in 1991 near the same time as proposals from Columbia and Sony emerged. The routing in the IBM proposal is based upon a rarely used IP option called *loose source routing* (LSSR). The entities involved with mobile communication include base stations (BASs), mobile routers (MRs), the MH itself, and a CH to correspond with the MH.

An important part of this proposal is the LSSR option. Source routing options allow the host sending the packet to dictate which routers it must traverse in order to reach its destination. In this way, the source can force a route that would not normally be taken (which is exactly what we want to do). Loose source routing still requires that the packet follow the designated path, but it also allows for multiple hops between routers on this path [Com91]. It would seem that this is a good alternative for mobility support, and in fact mobility support using LSSR-like options in IPv6 are integrated into a current Mobile IPv6 proposal [PJ96].

The idea of a home network for the mobile host is best represented in the proposal by the MRs which claim reachability to a set of addresses corresponding to mobile hosts. An MH may have one or many MRs advertising reachability to the MH. Since all packets have to pass through an MR unless two MHs are on the same network, essentially the mobile host is never home.

When the mobile host attaches to a foreign network, it must register with a *Base Station* (*BAS*) on that network. Once it has registered, the mobile host must inform its home router (MR) of the home address belonging to the BAS to which it has registered. The home router itself sends reachability messages about its MHs to nearby routers so that the packets get directed to it. An MR can also be a BAS, and this simplifies routing when an MH is on the same network as its home router.

When a correspondent host sends a packet to the MH, it is directed to the MR. This MR routes the packet through the appropriate BAS using LSSR. When the BAS receives the packet it will route it to the MH. Any reply messages sent from the MH to the CH will follow the reverse LSSR route using the same IP option. The next packet sent from the CH to the MH should also have an LSSR route attached to it (in response to the LSSR option from the MH), and therefore

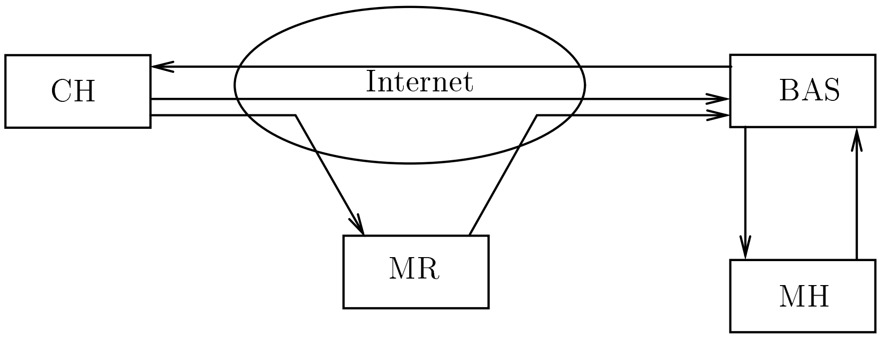


Figure 6: Routing in the IBM Proposal

the optimal route bypassing the MR will always be used. Figure 6 shows routing in the IBM system.

When an MH moves from one foreign network to another it must register with a new BAS. Once registered, the MH must notify its home router of its location, and inform the previous BAS of the location change. When a packet is directed at the previous BAS, this BAS routes the packet to the MR which routes it to the correct destination. Any return packets to the CH will contain the new loose source route, and the previous BAS will no longer be used for routing between these hosts.

The main problem with this protocol is its use of loose source routing. Like other IP options, LSSR is not always implemented, and when it is, it is not always implemented correctly. The initial IBM proposals, however, were flagged by Andrew Myles as one of the most promising for a long term solution to mobility support [MS93b]. Charles Perkins noted the problems with the protocol, and used his experience to help develop subsequent protocols with Andrew Myles and David Johnson. The most recent of these protocols is introduced in the next section.

6.4 The Internet Mobile Host Protocol: IMHP

The IMHP proposal resulted from collaboration amongst Andrew Myles, Charles Perkins, and David Johnson with respect to the Columbia, Sony, IBM and other proposals. IMHP was released in 1994 concurrent with Harvard Mobile IP and the beginnings of the IETF discussions.

IMHP is based upon four entities: a mobile host (MH), a local agent (LA), a cache agent (CA), and a home agent (HA). The mobile host is as defined in other proposals, and will not be redefined here. Descriptions of the other entities will be given below.

A local agent (LA) is a mobile routing agent on a foreign network to which the MH connects. When connecting to a foreign network, the MH must register with an LA connected on the same foreign network. Each LA maintains a list of MHs visiting its network, and uses this list to resolve local network addresses from the IP addresses of the MHs. Visitor list entries are controlled by time-outs and the MH must periodically re-apply to maintain connectivity. When the MH registers, the LA assigns it a care-of address to which the packets from the home agent will be forwarded. Care-of addresses are generally one of the local agent's addresses. The registration procedure is completed when the previous local agent is informed of the move, and it deletes its mobility binding from the visitor list.

Cache agents (CA) are IMHP-aware hosts which can cache care-of address bindings for mobile hosts. The IMHP protocol distributes the location information for the MH as needed to CAs throughout the Internet. If a cache agent receives packets addressed for an MH in its binding cache, the CA will encapsulate the packet and forward it to the care-of address specified in the binding. Location cache entries are controlled by time-outs, and a cache agent can attempt to re-confirm an MH's location before it times out in order to retain valid information. Local agents can also be cache agents, and upon being informed of movements from their network can create a location binding for the MH when the visitor binding is deleted.

The home agent (HA) is a permanent cache agent and possible local agent for an MH whose network address corresponds to the HA's. The HA will act as a local agent when the MH is on its home network. When acting as a local agent, the HA will place a binding for the MH in its visitor list. This binding will never time out and will only be deleted when the HA receives notification of a move.

When the MH is located on a foreign network the HA acts as a cache agent. Normal IP routing will always route packets for the MH to its home network unless they are intercepted. For this reason, the HA should always be able to intercept packets as a last resort in the routing. When packets arrive on the home network, the HA will intercept them, encapsulate them, and forward them to the MH's care-of address. If the cache entry on the HA times out then the HA will assume that the MH is on the home network. Details of routing for the various agents will be given in Sections 6.4.3, 6.4.1, and 6.4.2.

The specification states that any tunnelling mechanism should work with IMHP. For our purposes, we can assume that standard IP-in-IP forwarding is used. IMHP does specify a specific type of tunnel, called a special tunnel, that is used to direct a packet to an MH on its home network in an encapsulated state. Essentially, a special tunnel is a mechanism for encapsulating packets for a certain destination in a tunnel destined for the same destination. This tunnel indicates erroneous operation in the protocol.

The IMHP also specifies a weak security model for the protocols. The weak security model specifies ways in which the MH, CAs, LAs and HA authenticate themselves to each other. This authentication involves exchanging random numbers and essential information. Authentication is needed to ensure that the agents are forwarding correct information to the MH, and that the information is being forwarded to the MH itself rather than an impostor. The security mechanism details will not be discussed here.

6.4.1 Cache Agent Rules

The cache agent is responsible for forwarding packets to the MH. Certain rules are needed to help a CA decide when to forward, when not to, and how to forward. The following rules describe forwarding in IMHP [JMP94]:

- If the CA receives a special tunnel packet it should forward it normally.
- If the CA receives a packet destined for an MH for which the CA has a binding should encapsulate the packet and forward it to the MH's care-of address.

- If the CA receives a packet tunnelled to itself, but has no binding for the MH destination, it should special tunnel the packet to the MH's home network.
- In all other cases normal routing should be used.

6.4.2 Local Agent Rules

The local agents exist on the same network as the visiting mobile host and are used for receiving forwarded packets and delivering them to the MH. The following rules apply to local agents [JMP94]:

- If the LA receives a special tunnel packet it should forward it normally.
- If the LA receives a tunnelled packet with itself as the destination, and it has a binding for the destination of the encapsulated packet, then it should decapsulate the packet and deliver it to the destination using the binding.
- If the LA receives a packet for a host for which it has a binding in its visitor list, it should deliver the packet on the local network.
- In all other cases normal routing should be used.

6.4.3 Home Agent Rules

The home agent is a special combination of LA and CA with extra responsibilities of its own. The following rules apply to HAs in IMHP [JMP94]:

- If the HA receives a special tunnel packet and it has a record of the destination host in its visitor list, it should decapsulate the packet and apply the following rules.
- If an HA receives a packet destined for an MH in its server list and it has a visitor list entry for that host, then the HA should forward the packet to the MH using the visitor entry information.
- An HA should never tunnel a packet back to a host that has just tunnelled it to the HA.
- If an HA receives a packet for an MH in its server list, but has no bindings for that MH in its location cache or visitor list cache, the HA should discard the packet.
- In other cases, normal IP routing should be used.

6.4.4 Notification Messages

In order to reinforce some of the rules for routing, and to let information about mobile hosts propagate through the internetwork, notification messages are needed for communication between the routing entities. The following notifications are needed for managing the IMHP system:

- A CA which encapsulates a packet should notify the source of any bindings the CA has for the MH destination.

- A CA or LA which cannot forward an encapsulated packet except by a special tunnel should inform the source of an invalid cache entry.
- An MH should inform its HA of all migrations.
- A CA must authenticate the MH before intercepting packets.

6.4.5 Summary

The IMHP proposal combined many ideas from previous proposals and produced a system which addressed many faults of those proposals and included security mechanisms for authenticating entities. In addition, notification messages between the entities involved in the mobile routing were simple and not prohibitive for implementation on the Internet. The development of this protocol addressed many concerns needed for the IETF proposal, and was a good basis for its development.

6.5 Harvard

In 1994, Harvard University developed and implemented a Mobile IP system for efficiently handling Internet connectivity when crossing wireless cells [BCC+94]. This system was developed after the Columbia, IBM, Sony, and even the early versions of the IETF proposals were released. As a result, this proposal addressed many concerns that were raised about the other proposals. Since the release of this Mobile IP system, some of the design innovations and improvements have been incorporated into the IETF Mobile IP specifications (see Section 7).

The basic routing proposed here is very similar to that of the IETF basic dog-leg routing. The name dog-leg routing comes from the indirect route through the home agent (HA) which is similar to bend in a dog's leg (see Section 5.6). In all routing schemes, the MH has a permanent home address on a home network. Located on this home network is a home agent (HA). When an MH is away from the home network, it is the HA's responsibility to redirect packets to the MH's location. A network other than the home network of an MH is called a foreign network. When the mobile host enters a foreign network it must find a foreign agent (FA) on that network in order to achieve connectivity to the Internet. It is the job of the foreign agent to accept packets from the HA destined to the MH and deliver them directly to the MH.

In the standard routing description, packets are sent by a correspondent host (CH) to the MH's home address, and therefore the MH's home network. The HA on the home network intercepts these packets, encapsulates them, and sends them to the FA. The FA decapsulates the packets, and forwards them directly to the MH using the appropriate network-level protocols.

The enhancements developed for this system include short-cut routing mechanisms and improved security mechanisms. The short-cut routing removes the dog-leg in the packet route by allowing the CHs to communicate directly with the FA involved. Since this requires notification of movement and security relationships for the CH, this extension greatly complicates the protocol.

The architecture for the Harvard system can be divided into four protocols: hand-off, registration, location update, and routing/encapsulation. Each protocol runs on multiple hosts in the system, and interfaces are designed for simplicity and ease of use. These protocols are briefly introduced in the following subsections.

6.5.1 Hand-off

Hand-offs are the part of the system dealing with MH movement across network boundaries. For the purpose of the Harvard proposal, these boundaries correspond to wireless cell boundaries. Since in many cases it would make sense for wireless cells to be subnets of a local network, these boundaries can correspond to address boundaries as well. If the wireless cell boundaries do not correspond to address boundaries, then a single foreign agent on the network should be able to handle forwarding regardless of which wireless cell the MH is in.

When an MH crosses a network address boundary, a new FA must be assigned for the MH. Each FA periodically broadcasts beacons. Upon receiving a beacon, an MH can register with the FA if it so desires (usually if it is not already registered). Registration causes default routing information to be stored in both the FA and the MH, and notification to be sent to the HA.

The MH is responsible for ensuring the registration, and must periodically tell the FA it still requires service. Time-outs remove routing information from previously used FAs if re-registration requests are not received.

6.5.2 Registration

Registration is the protocol by which an MH informs its HA of location changes. Registration serves two functions: security and routing accuracy. An MH registers by sending its IP address and its FA's IP address. Security messages involving random and known security numbers are exchanged to authenticate both the HA and the MH. This is needed to guard against hosts pretending to be either of the two entities involved in the transaction. Once the MH has been registered, correct forwarding of packets can begin. The Harvard system requires the MH to periodically re-register to guard against data loss at the HA due to reboots. Each re-registration requires a different security code to guard against replay attacks by malicious agents (See [Atk94] for information on replay attacks).

6.5.3 Location Updates

Short-cut routing involves allowing CHs to directly communicate with MHs. This is facilitated by having the HA inform any CHs using dog-leg routing that the MH is mobile. Upon receiving notification of mobility, the CH will apply for a *subscription* to the HA. Before the CH does this, however, it must authenticate the HA. It does this by asking the MH in a secure fashion what the address of its HA is. If the HA address checks out, then the CH applies for a subscription with the HA.

Subscriptions ensure that CHs will be notified by the HA of all moves that the MH makes. In order to reduce the number of subscriptions active, there are time-outs in effect and the CH must re-subscribe periodically in order to keep the subscription active.

6.5.4 Routing and Encapsulation

The encapsulation used in this proposal is essentially IP-in-IP (see Section 5.3). A special IP protocol number in the outer header identifies the packet as encapsulated. FAs identify packets they should forward by the destination address in the inner packet. These same FAs silently discard any packet destined for MHs they do not support. FAs may forward packets to new FAs if the new care-of address for the MH is known. Time-outs at higher levels let the CH know if the packet was delivered or not.

6.5.5 System Performance

The performance of the Harvard proposal has been evaluated by examining TCP throughput, TCP delay during handoffs, and registration speed. The results seem to show that there are definite performance gains to be had by using the short-cut routing. Dog-leg routing yielded a performance of 1.1 Mbps on their testbed, while short-cut routing yielded 1.3 Mbps. Routing performance across the wireless link itself was measured at 1.3 Mbps. This indicates performance gains can be had through short-cut routing, however, all routing was done on a single ethernet, and more extensive tests need to be performed to validate these results.

Hand-off time affects TCP delay in a stepped fashion due to TCP's time-out mechanisms. The results show that hand-off delays < 1.2 seconds generate a 1.2 second delay at the TCP level, while hand-off delays between 1.2 and 3.4 (delays between 3.4 and 4.4 were not tested) will generate TCP delays around 4.4 seconds. These results show that small delays during hand-offs can be greatly exaggerated at the TCP level.

Registration measurements showed a large capacity for registrations with this system. Each registration takes about 20 milliseconds including transmission time. This would allow up to 54 registrations per second. These results were obtained with all hosts on the same ethernet. Actual registration times could be much higher over the Internet, however registration capacity at the HA should still be high.

6.5.6 Summary

The Harvard protocol is a robust design and has many advantages. There are working implementations of the design, and their performance in many respects seems to be quite comparable to fixed IP. One potential problem with the system is the large number of messages which need to be exchanged in order to register, handle security, and update MH locations. Although this type of interaction is common in many of the modern IP systems, its necessity (especially with respect to short cut routing) is still in question.

7 The IETF Mobile IP Specifications

The Internet Engineering Task Force (IETF) is a collection of researchers organized into working groups for the purpose of exploring issues related to the Internet. The IETF has been around since the Internet research organizations were reorganized by DARPA in 1983. IETF members are volunteer researchers, and are not funded by the IETF.

The purpose of the IETF is to study engineering problems in the Internet, propose solutions to those problems, and evaluate the proposed solutions. Internet proposals are usually presented first as Internet drafts. These drafts are relatively informal documents that outline a proposed solution. Internet drafts are submitted to working groups usually by e-mail to the appropriate working group mailing list. Drafts are discussed on these lists and in yearly IETF meetings. Drafts are revised, discarded, or otherwise treated accordingly to their merit, and eventually may reach RFC (request for comment) status. An RFC document is essentially the result after a draft has been examined and approved by the working group involved. RFCs are standards documents stating a definitive standard for the Internet, and they can be a more formal presentation of a proposed solution. All RFCs are numbered and are available freely on the Internet.

This section will present the Mobile IP architecture which has been developed by the IETF. Many of the research groups discussed in previous sections have made contributions to this proposal and its features have come from the previous proposals and discussions within the group. The Mobile IP specification is currently on its 17th draft iteration, and is close to being accepted as an RFC. IETF Mobile IP will be introduced by examining the architectural entities which make the protocol work, this is followed by a discussion of the services provided by Mobile IP, and finally, the routing and routing enhancements being proposed for IETF Mobile IP will be discussed.

7.1 Architectural Entities

The IETF Mobile IP proposal is based on three main entities: a mobile host (MH), a home agent (HA) and a foreign agent (FA). These entities operate on different networks called the home network (HN) and foreign network (FN). Hosts exchanging packets with MHs are called correspondent hosts (CHs). Some of these entities are already familiar from other proposals, but they will be discussed here in terms of the IETF model.

A mobile host (MH) is an entity which changes its point of attachment to an internetwork from one network or subnet to another. An MH has associated with it a permanent address which acts as an identifier for the host. Connectivity should be maintained while possible through this address regardless of movement.

A portion of every Internet address is known as the network/subnetwork component. This component is used when routing data to a host. In the permanent identifier for an MH, the network component identifies the MH's home network. All packets to the MH are routed to this network by default IP routing mechanisms.

When MHs roam the internetwork they must temporarily connect to networks other than their home network. These other networks are called foreign networks. When an MH connects to a foreign network, it is required to register with a foreign agent (FA). It is the job of the FA to act as the default router for the MH. This requires accepting encapsulated packets on behalf of the MH and delivering these packets directly to the MH.

The home agent (HA) is a host on the home network that is responsible for maintaining connectivity for a group of MHs. When an MH is away from its home network it will inform the HA of its location and the address of an FA acting on the MH's behalf. Once an MH has registered, the HA is responsible for telling all other hosts on the home network that it will handle the MH's packets. Once the registration has proceeded, and the home network routing is set up, the HA can begin intercepting packets bound for the MH on the home network. These packets are encapsulated using IP-in-IP and forwarded to the FA. In addition to forwarding packets, the HA is responsible for maintaining security information about the MH and forcing location updates to be authenticated using this information.

Table 2: Agent Advertisement Fields

Field	Description
Type	16 (the number for router advertisements)
Length	The length is (6+4*N) where N is the number of care-of addresses advertised.
Sequence number	Count of advertisements sent since agent started.
Registration lifetime	Maximum length registration acceptable
R	Registration required, no co-located addresses offered.
B	Busy, will not accept registrations.
H	Home agent services offered.
F	Foreign agent services offered.
M	Minimal encapsulation supported.
G	GRE method encapsulation supported.
V	Van Jacobson header compression supported.
Reserved	Reserved for future use.
Care-of addresses	A list of address MHs may use on this network.

7.2 Support Services

In order to handle mobility, an internetwork needs to support a host when it is connected and when it is in transition. While connected, the IP routing mechanisms enhanced with the Mobile IP agents handle communication effectively. Connection-state changes require additional support outside of the routing domain. This support involves tracking new locations, keeping track of agent states, and reconfiguring the protocol when changes occur. Connection-state support is handled in Mobile IP by *agent discovery* mechanisms and *registration services*. This section will discuss both of these services.

Agent discovery supports communication between the MH and agents on a network to which it is connected. Mobility supporting agents (HAs and FAs) periodically broadcast *ICMP router advertisement messages* which contain mobility information. When an MH receives an advertisement message, it either updates the lifetime of its foreign agent entry, confirms its location, or recognizes that the router message has arrived from a network different from the one it is registered to. Since wireless connections are possible, receiving agent advertisements from a new network does not mean that the host cannot communicate with its previously registered FA. If the MH does receive advertisement messages from a new network, and the old FA binding times out, then a network movement is indicated, and registration procedures must proceed on the new network.

Agent advertisement messages contain many fields, which are shown in Table 2. Two important fields are registration lifetime and care-of addresses. The registration lifetime sent in the message is the maximum registration time that the agent is willing to support. These are the lifetimes that are used to time-out entries in the FA's visitor list and invalidate FA bindings at the MH.

Care-of addresses are identifiers on the agent's subnet that can be used by the MH for connectivity to the internetwork. There are two types of care-of addresses: co-located, and agent-located. Co-located addresses are temporary identifiers for the MH itself on the foreign network. These identifiers would be taken from a sequence of IP addresses reserved for temporary use. Agent-located addresses are simply the IP address of the foreign agent itself. In this case, the agent will receive packets destined for the MH, decapsulate them, and forward them on the foreign network. Each foreign agent must advertise at least one care-of address, but may advertise more.

If an MH is not receiving agent advertisements, or if its registration has timed-out, it can begin *agent solicitation*. Agent solicitation messages are special ICMP router solicitation messages and follow the same rules and procedures. All agents must respond to agent solicitations on their subnet. An agent is required to respond with an *agent advertisement message*.

An MH needs to detect whether it is on a foreign network or its home network. This is can also be done through agent advertisements. One of the fields in the advertisement messages is used to specify if the host is a home agent or foreign agent. If the host is the home agent for the MH, then it must treat the first HA message as a special case. While the MH is on its home network, normal IP routing is sufficient for routing its packets, and certain measures have to be taken to ensure Mobile IP does not interfere.

When an MH detects its home network, it needs to reconfigure its routing tables so that hosts on the local network can be reached directly. Once this is done, the MH must deregister with its home agent. Deregistration causes the HA to update the MH's location information in its MH support list, and adjust its routing tables to reflect the MH's new location. When these two steps have been taken, the MH will function as a normal fixed host, and the HA will no longer intercept packets on behalf of the MH.

When an MH detects that it is on a foreign network, the MH needs to inform a foreign agent of its presence and register with an HA. A foreign agent can refuse to service the MH if it is too busy. In this case, the MH must wait until it receives an agent advertisement message stating that the foreign agent is willing to host the MH. The HA must always be willing to register an MH that it services.

Registration provides the MH with the mechanisms for communicating its reachability information to the important routing entities. The use of the following registration mechanisms are required for Mobile IP to operate properly:

- Request service for packet forwarding when connecting to a foreign network.
- Keep HA informed of current care-of address.
- Renew registrations to prevent binding expiration.
- Deregister with HA upon connection to HN.

In addition to these basic capabilities, the following optional support capabilities are provided for tracking MHs:

- Multiple registrations can be maintained simultaneously by an MH. This results in duplicate packets being tunnelled and delivered to the MH via different foreign agents. When using wireless networks, this reduces packet loss during hand-offs and increases the robustness of the system.

- Subsets of the mobility bindings held by the MH can be deregistered. When multiple bindings are held, this is useful to deselect bindings over wireless interfaces for which the signal strength has become too low.
- An HA's address can be discovered dynamically if the MH is not configured with one. This capability can be implemented by broadcasting registration requests onto the home network. All HAs on the network will respond with packets denying the registration, and the MH can use the source addresses in these packets to re-register with an individual HA.

The mobile host can register through an FA or directly with its HA depending on the circumstance regarding its registration. If an MH is using its FA's address as a care-of address, then it must register through that FA. If the MH is using a co-located address, it can register through the FA, or send registration messages directly to the HA itself (an FA can force registration through it by setting a bit in the agent advertisement message). Registration through an FA involves directing the registration towards the FA which then forwards it to the HA. This may be required for effective tracking of FA usage on the foreign network. If a mobile host returns to the home network, it must register directly with the HA. The proposal specifies that registration messages should be sent via UDP, which may necessitate multiple registration requests being sent due to packet loss.

7.3 Routing and Routing Enhancements

The basic routing in the IETF proposal is the most conservative and involves at least four entities (MH, CH, HA, FA). Packets are intercepted on the home network by informing all hosts on that network that the MH has relocated and the HA is acting on its behalf. A common approach for doing this on an ethernet is called *proxy ARP*. The abbreviation *ARP* stands for *address resolution protocol*. ARP is a protocol by which a host can ask for a mapping from a local IP address to a local link-level address for some host on the network. The appropriate host is required to reply with the desired mapping. These addresses are used for forwarding packets on the local network, since the hardware usually does not operate with IP addresses. The protocol also allows a host to broadcast the mapping from its IP address to its link-level address.

Proxy ARP is a refinement by which a host can broadcast a mapping from another host's IP address to the first host's link-level address. In this way, packets destined for the other host will be forwarded to the first host on the local network. This is useful in Mobile IP. Suppose the MH's IP address is 1.2.3.4 and its link-level address is 1.2.3.4.5.6. Lets also say the HA's link-level address is 6.7.8.9.10.11. If the HA broadcasts a proxy ARP message saying that 1.2.3.4 maps to 6.7.8.9.10.11, then all packets destined for 1.2.3.4 get delivered to the HA [Com91].

Two extensions to basic routing include optimized routing, and multicast routing. Both are discussed briefly in the IETF proposal, but the details are left as extensions to the proposal, and have been developed separately. Secure short-cut routing attempts to address any routing inefficiencies that may result from having the default dog-leg routing and remove the dependency on the home agent as the sole method of routing a packet to the MH. Short-cut routing proposals attempt to eliminate the HA from most communications and allow the correspondent hosts to communicate directly with any MHs it may be in contact with. Multicast routing extensions allow the MH to receive multicast packets while away from its home network. Mobile multicast routing was introduced in Section 5.5. These extensions will be described in the following subsections.

7.3.1 Optimized Routing

Optimized routing includes extensions to the basic IETF proposal which try to improve the reliability and efficiency of the mobile routing. The most significant research has been with respect to short-cut routing protocols. *Secure short-cut routing* was first proposed by Charles Perkins in 1992 (see [PR92]). A version of route optimization for Mobile IP has been recently proposed by David Johnson [JP96]. Although the end-to-end performance benefits are open for debate, it seems as if short-cut routing is an extension which is desirable for adequate testing of proposals being made for IPv6, and for maintaining hop counts at a reasonably low level even in the presence of mobile hosts.

Optimized routing is designed to provide multiple benefits with regards to packet routing to the MH. These benefits include reducing the cases where the dog-leg is followed for routing, but also includes methods for ensuring packet delivery during movement on the net. These benefits arise by not only allowing the HA to cache the location of the MH, but also allowing CHs to cache location information. The argument in favour of short-cut routing is that it helps to avoid situations where the CH and MH are very close in terms of physical networks as compared to where the HA is in relation to either. In this case, the distance travelled can be increased many times over by the dog-leg route through the HA.

Problems with short-cut routing include the increased overhead involved with keeping the additional routing agents up-to-date, and the security overhead required to ensure that all agents handling mobile IP routing are trusted. It is debatable whether the benefits are worth the additional overhead.

In order to specify short-cut routing, additional terminology beyond that for the basic protocol is needed. A *binding cache* is defined to be a cache of location information for mobile hosts which is held at a correspondent host. A binding update is a notification message used to update information in binding caches. *Special tunnels* (as described in Section 6.4) are tunnels with the outer address being the MHs home address. Special tunnels allow CHs to force a packet to reach the home network, and therefore the HA.

Short-cut routing in Mobile IP can involve many entities. We will begin with the foreign agents that an MH may encounter. When an MH moves to a different network, it will usually deregister with one or more foreign agents. In the original system, the foreign agent will forget about the MH. Short-cut routing requires the MH to inform the old FA of its new care-of binding. The old FA will then create a binding cache entry for the MH and will forward any packets that are mistakenly delivered to the previous network.

As mentioned above, previously used foreign agents can receive datagrams destined for an MH which has moved on to other networks. This situation indicates that a CH or other host with a binding cache entry for the MH has invalid information. If the old FA has a binding cache entry for the MH, then it should forward the packet to the new FA. The old FA can also send notification to the HA via a *binding warning message* to inform it that an Internet host has an out-of-date binding cache entry for the MH. It is the HA's responsibility to send a binding cache update message to the offending host to update its cache. If an FA receives an encapsulated packet addressed to itself, but does not have the MH identified in the inner header in its visitor list, then that FA should send the packet to the HA via a special tunnel. MHs can keep their previous FAs informed of their location by requesting their current FA to inform the previous FA via a *binding update* message. This is done during the registration procedure with the FA.

The purpose of short-cut routing is to reduce the involvement of the HA in the transmission of packets to the MH. However, because the routing is now spread to so many entities, the HA has more responsibilities. When normal dog-leg routing is being used, it is the job of the HA to route the packets to the MH's care-of address. With short-cut routing, the HA should also send a binding cache update message to the sender of the packet to inform it that the MH is away from its home network. If the sender can support a binding cache, it should create an entry for the MH upon receiving an update. The HA continues to send binding cache updates as packets arrive, but will quickly back off and stop sending after the first few packets. Additional updates are sent in case the first few are lost, however updates are no longer sent once it is obvious that the CH does not support mobility forwarding mechanisms. If the HA receives binding warning messages, it is required to send a binding update message to the host indicated in the warning.

Correspondent hosts can use short-cut routing to improve the efficiency of their transmissions to the MH. The binding cache entries at the CH also allow it to act as a router for the MH when it encounters packets destined for the MH. This may reduce the path length that the packet has to take if it is intercepted before the home network. A CH is informed of an MH's location via binding cache update messages from the HA. If the CH supports a binding cache, then it will create an entry for the MH and cache its location. The binding cache entry is assigned a lifetime equal to the lifetime remaining in the MH's registration with its current FA. Once a binding cache entry times out, the CH must remove it from the cache. CHs desiring to maintain the cache entry can request a binding update from the HA, which will send the current location information for the MH if it is available. A CH with a binding cache entry is allowed to intercept packets destined for the MH via its home address. The CH should then encapsulate the packet and forward it as the HA would. If the CH forwards the packet to the wrong location, then it should eventually be informed via binding cache update messages of the error and any corrections necessary.

The ability of cache agents to intercept packets and forward as HAs would creates many security risks. These risks and others are discussed in Section 7.3.3. The bottom line with these risks is that they require mechanisms to support security. Mechanisms need to be chosen which do not cost more in performance than the short-cut routing is worth.

Research has been performed to examine the costs of route optimization in terms of update overhead and other factors. This research indicates that the update overhead is prohibitive if the ratio of location changes to connections is high, and basic routing is more desirable in this case. If the ratio of location changes to connections is low, then updates are more efficient than the basic protocol. An adaptive location management scheme has been developed which adjusts the degree of route optimization according to the characteristics of the traffic and the host movement [RB95].

At the IETF meeting in June, 1996, proposed changes to the route optimization draft were outlined. Essentially they included making the route optimization in IPv4 work like it does in IPv6 (see Section 11.1.1) [Sol96].

7.3.2 Mobile Multicast Message Delivery

Multicasting is a method by which packets can be efficiently delivered to a group of hosts which reside on many different networks in an internetwork [DC90]. Multicasting is very useful in teleconferencing applications, and is the mode of communication for the Mbone [Eri94]. The main idea behind multicasting is that a sender can send out a single packet to a multicast address. The routers in the Internet hold information for different multicast groups which indicate if they are to forward the packets, and if so where they should be forwarded to. The power of multicasting is that routers away from the original source can duplicate the packets and send them on multiple interfaces. In this way, traffic can be conserved by transmitting single packets until they have to be duplicated, and trying to duplicate them as close to the destination as possible. The techniques for duplicating these packets have been studied extensively. See [DW88], [Dee89], [DEF+94], [Moy94], and [Dee89] for more information on multicasting and its mechanisms.

Mobility and multicasting together create a problem. The problem is that efficiency is gained through multicasting by maintaining a reasonably static delivery tree. Routers in the interior of the tree use information about the tree structure to direct packets to the hosts in the multicast group. Changes to the multicast group are expected to be infrequent, but are expected to be centered around changing membership, not changing location. Mobility introduces location changes to the multicast problem.

Multicasting is dependent upon multicast routers. These routers are widely dispersed, but are not everywhere. Mobile multicast support, therefore, cannot rely upon the MH having a multicast router to service it. Even if a multicast router were available, reconfiguring the tree would have to be done by the MH unsubscribing at its old address and resubscribing using its care-of address. Unless there are already hosts at the network subscribing to the multicast group, this type of restructuring takes some time and definitely would not be transparent to the user. As well, the overhead associated with frequent moves would be prohibitive to the network because of all the router updates required to change the multicast tree. A new approach independent of changes to the multicast tree is needed.

A simple solution to the multicast problem involves having the HA intercept the packets on the home network and forward them as normal to the MH. This will work fine for incoming packets, but outgoing packets will suffer because of built-in mechanisms with multicasting protocols which cause packets to be dropped if the path up the multicast tree differs from the path down the multicast tree used by the routers to send packets to the MH. Therefore, a bi-directional tunnelling mechanisms has been recommended for multicast delivery.

The IETF proposal actually allows both tunnels to the HA, and local subscription to the multicast datagrams. If local subscription is used, then the local care-of address for the MH must be used for the source address of all correspondent packets onto the multicast tree. If the MH does not use the proper address, then the multicast packets will likely be dropped. If the MH is using a bi-directional tunnel to its HA, then the MH must use its home address as the source address for all packets.

A detailed design for multicast support is given in [CWBM96]. This design is based on routing multicast packets through the home agent. Some extensions and modifications from the basic proposal are discussed in this paper. The following points are raised with respect to multicast routing:

- The foreign agent should multicast the packets onto the foreign network using link-level multicast, not unicast them to each individual MH.

- To avoid duplicate packet deliveries to the same FA, the FA should designate one HA to be the router for the multicast group (if there is not already a host subscribed to the group on the local network). The designated host will forward the multicast datagrams until it no longer has MHs on the network, or there are no longer MHs subscribing to the multicast group on the FN.

- Hosts offering services as foreign agents should also be multicast routers in order to detect the state of host subscriptions on the network.
- Time-To-Live fields can be used to scope multicast messages by ensuring that the packets will time-out before they leave the network. In order to handle local area multicasts scoped in this manner, the local area multicasts should be changed to have network-specific fields in their addresses.

7.3.3 Security Considerations

Security in the Mobile IP proposal is a big issue, and has generated a lot of concern. In general, the Internet itself is not very secure, and there are many threats to hosts. Mechanisms already exists to protect information on the Internet, and Mobile IP is usable with most of these methods (although firewalls present a problem due to their resistance to encapsulated packets and proxy ARPS). The problems are in ensuring that a foreign agent is really servicing the MH it says it is, that the host being serviced is in fact the MH it claims to be, and that any HA spreading information on the Internet about an MH is in fact the correct and justified host for doing so.

In the basic IETF proposal, the main security relationship needed is between the MH and the HA. It is recommended that certain security measures are hardcoded into an MH and its HA during initial configuration. These measures could be as simple as security keys that are known only to these two hosts. Through the use of this security mechanism, the MH can authenticate its HA, and an HA can confirm any MH it is corresponding with. A situation in which two mobility-aware hosts have a mechanism for authenticating themselves with each other is called a mobility security association (MSA). This MSA between the MH and the HA can be used to create an MSA for any foreign agent that the mobile host wishes to use via exchanges between the FA, the MH and the HA. Under the basic IETF proposal, this security required is relatively simple, it becomes more complex when short-cut routing is introduced.

With short-cut routing, not only the MH, HA, and current FA need MSAs, but as well, many routing agents or CHs on the Internet will require them. This presents a problem as the overhead with creating and maintaining MSAs for so many entities becomes overwhelming. Since there are no general authentication mechanisms or key distributions in the Internet today, manually configured MSAs must be used for CHs supporting route optimization. This means that any route optimization is restricted to only hosts which the MH is known to correspond with and will therefore be set up with an appropriate MSA. The security concerns in the IETF proposal greatly limit the power of the route optimization extension.

8 Implementations of Mobile IP

Since the IETF started developing its proposals there has been interest in designing and implementing Mobile IP. Over the last few years various groups have implemented Mobile IP. These implementations (either full or partial) have led to some changes in the IETF Mobile IP specification itself. Most of the current implementations are based upon IETF draft versions 15 or

16. The current version of the draft is undergoing the approval process for becoming an RFC. Many of the implementations have been for the linux platform. This is not surprising since linux is freely available, the source code is easy to access, and the implementation is new, therefore less cluttered with legacy code. Implementations are also out for BSD UNIX 4.3, Free BSD and others. This section will discuss some of the implementations with respect to where the functionality is implemented (i.e., in the kernel or outside), and what analysis has been done using the implementations.

8.1 Linux

The first complete version of Mobile IP for the linux operating system was developed at the State University of New York at Binghamton[DGL95]. Versions of this software have been released throughout 1995 and 1996. The current version is compatible with IETF draft version 16. The implementation uses daemon processes to handle the mobility support, and uses built-in tunnelling support for handling the packet forwarding.

Two daemons are used to implement this system, an agent daemon and a MH daemon. The first daemon handles the functionality associated with FAs and HAs. The agent requires two files which contain the visitor cache and the MH support information required by an HA. Both of these files, **mip-mh.ok** and **mip-vmh.ok** are stored in the **/etc** directory and should be protected by requiring root permission for all file operations. This implementation has been used at SUNY, Binghamton, and by Dr. Arup Acharya at NEC, Princeton. Early versions of the system were demonstrated at MobiCom'95.

Use of this implementation has shown some interesting side effects of mobility. The most significant of these is related to the back-off mechanism built into TCP. When packets are lost, TCP interprets the cause as congestion, and immediately backs off. When a mobile host changes network attachments, there are packets lost, and this back-off mechanism kicks in. Unfortunately, this results in a needlessly small amount of the bandwidth getting used when communication resumes [DGL95].

8.2 University of Stockholm

The University of Stockholm and Anders Klements have produced a mobile IP implementation compatible with IETF draft version 11. Although the implementation is in SUN OS 3.1.4, it has been designed with portability in mind and has subsequently been updated and ported to mach and other operating systems. This implementation is mostly encapsulated in a daemon process which handles much of the external protocol work. Key management aspects of Mobile IP and the parts of the protocol which are natural to embed in the IP routing code have been included in the kernel. In particular, a pseudo kernel device driver has been implemented in the kernel to handle mobile communications.

The pseudo device driver is handled as a network interface in the kernel called mip0. When mobile messages are generated by an agent, they must be sent to the device driver interface which will handle the job of encapsulating the packets. The driver supports both IP-in-IP encapsulation and minimal encapsulation. Minimal encapsulation is a form of encapsulation which reduces the load on wireless links by decreasing the header size.

The latest version of the implementation is compatible with IETF draft 14. This version has been running in a test environment, and performance analysis has been done with respect to latency and throughput.

8.3 University of Saskatchewan

The University of Saskatchewan has implemented the basic routing for mobile IP on a 4.3 BSD UNIX system [CWB96]. The group is currently working on porting the implementation to HPUX 9.0, and adding multicast functionality according to their specifications. Preliminary testing has been done on the protocols, but performance results are not yet available for either the base or multicast implementations.

8.4 University of Singapore

The University of Singapore has developed an implementation of Mobile IP which exists in the linux kernel, and is compatible with IETF draft version 15 [LCT95]. The first version of this implementation was released in 1995, but has been further developed since. The latest version was just released in July, and is the fourth version of this implementation. The implementation claims full compatibility with IETF 15, as well as the short-cut route extensions defined by David Johnson. The latest version includes bi-directional tunnelling to the HA, and addresses the home network edge routing problem. As was mentioned in the IETF proposal, bi-directional tunnelling to the HA is essential for some implementations of multicast. Bi-directional tunnelling also helps with firewalls, as the tunnelled traffic is allowed through when it is bi-directional.

In July of 1996 this implementation was tested using an implementation in Europe to exchange packets with. The tests revealed that the two implementations were fully interoperable, and that the mobile IP was implemented correctly. The group has not reported any findings with regards to performance analysis, however [Lia96].

8.5 Other Implementations

Versions of Mobile IP have also been implemented by several other organizations. In particular, CMU has implemented Mobile IP under the direction of David Johnson and Charles Perkins has supervised an implementation of Mobile IP at IBM. Other organizations including FTP Software, Motorola, Nokia, SUN, and Telxon have also implemented the protocols [Sol95]. More implementations are appearing at a rapid pace, and Mobile IP appears to be the mobility protocol of preference for the global Internet.

9 Current Projects in Mobile Networking

This section will present some of the larger research groups exploring mobility and Mobile IP. Although the research performed by these groups is not restricted to Mobile IP protocols, many of them are addressing mobility concerns which will be impacted upon by Mobile IP. Some of the groups are studying higher-layer protocol interaction with Mobile IP and mobility in general.

9.1 DATAMAN

The DATAMAN project is based at Rutgers university. This project involves research in various aspects of computer mobility. This project group is experimenting with mobile computers running the protocols for Columbia University's Mobile IP specification (see Section 6.1). One of this project's significant contributions is the Indirect TCP (I-TCP) protocol [BB94]. Current research is being conducted into providing environment-aware support for many popular Internet services such as e-mail, ftp, and http.

The DATAMAN project has two of the major players in Mobile IP development (Tomasz Imielinski and B. R. Badrinath) who created the Colombia Mobile IP standard and have authored various RFCs for the IETF working groups. The DATAMAN group has developed interests in many aspects of mobile computing, and these are outlined below:

- Distributed algorithms and services
- Data broadcasting
- Indirect protocols
- Location management and data replication
- Multicasting and mobility
- Software architectures and programming support for mobile computing

The DATAMAN group is continuing to analyze mobility, but they are focusing upon mainly the Columbia implementations which are limited in terms of Internet-wide applicability [Bad96].

9.2 Monarch

The Monarch research group is headed by David B. Johnson at Carnegie Mellon University. David B. Johnson has been very active in Mobile IP development and is currently investigating mobility support from the link-layer through to the presentation layer.

The goal is to develop a seamless mobile working environment for the end user. The scope of their research includes protocol design, implementation, performance evaluation, and usage-based validation. Recent papers from the group deal with mobile multicast [JM96b], wireless infrastructures [HJ96], and dynamic source routing [JM96a].

9.3 University of Singapore

The University of Singapore deserves special mention because of the variety and quantity of research underway relating to IETF Mobile IP. Graduate students at Singapore are studying Mobile IP at a variety of levels. Some of the students are developing mathematical models to represent Mobile IP behaviour. These models are used to analytically predict performance from Mobile IP. Other researchers are developing simulation models of the protocols. These simulations will be used to conduct experiments not possible in the current Internet. Another set of graduate students are implementing Mobile IP and performing analysis on their implementation. This analysis can be used to lend support to or dispute the validity of the analytical and simulation models.

9.4 Other Research

Many other groups are researching mobility at the IP level, below it, and above it. Groups like AMIGOS [Ami] are taking a comprehensive view of mobility and trying to design entire application systems with the IP mobility built-in. MosquitoNet is a research project at Stanford which is studying methods of improving applications so that they can effectively handle the performance variability associated with mobility. Their testbed includes an implementation of Mobile IP to study transmission characteristics using this protocol [BZCS96]. Other researchers are concentrating on improvements to the IETF proposals. Current work is under way regarding the forwarding of packets from one foreign host onto the next so that as an MH moves it leaves behind a trail of foreign agents which can forward the packets along the MH's trail more efficiently than what would result from router updates and HA deregistration with every move. Other researchers are examining packet duplication and TCP performance enhancement at the FAs [HBK95].

Users commonly use a specific set of protocols which rely on IP to transmit their data. Many of these protocols (such as TCP) are designed to interpret events such as packet loss as indications of an Internet condition. TCP, for example, treats packet loss as a sign of congestion on the Internet and backs off its transmission rate.

With the advent of Mobile IP and the emergence of implementations, the reaction of higher-layer protocols to location changes can be studied in a similar manner. Some preliminary studies have already been performed with the TCP protocol. The results indicate that TCP will react badly to packet losses resulting from location change. To counteract this, either a new version of TCP is required, or the protocols and possibly hardware at layers below TCP must be changed to provide TCP with the type of performance that it expects.

A new version of TCP has been proposed to help with packet loss problems over wireless interfaces. This solution involves having the wireless connection to the fixed network repeatedly broadcast and accept acknowledgements of packets without involving the source for the packet. In this manner, packets which reach the wireless interface can be acknowledged by the wireless interface itself which will guarantee delivery. This shields TCP operation from packet loss with a wireless network [BB94].

A variation of Mobile IP has also been proposed to help with TCP interaction [HBK95]. This proposal defines mechanisms for a Mobile IP protocol which will allow multiple foreign agent caching and duplicate packet delivery in order to stop packet loss with transitions across network. A key part to this protocol is that the foreign agents actually cache packets destined to the MH, and when a transfer of management occurs, the new foreign agent already has packets to forward which may correspond to ones lost in the transfer.

Other research in the area of mobility is proceeding rapidly. Research groups are studying mobility with respect to ATM, the Web, databases, and many other facets of computing. One area of interest right now is the next generation of IP protocols. Research groups have taken the knowledge already gained from Mobile IP implementations in IPv4, and are using it to design the mobility support for IPv6. The following section will discuss IPv6 and its mobility support proposals.

10 The Next Generation of IP

10.1 Introduction to IPv6

TCP/IP was first proposed in 1974 by Cerf and Kahn [CK74] for routing in an internetwork composed of heterogeneous networks, and ensuring end-to-end interprocess communication. This work was the basis of Cerf's Ph.D. thesis at Stanford University, and he was surprised when the defence department wanted to adopt it for the Internet. In retrospect, Cerf has said that he would have given TCP/IP a larger address space and made other changes had he known at the time of its development what it would be used for. The full specification for the IP Internet protocol can be found in RFC791 [JP81a]. The TCP/IP suite of protocols grew and expanded, and quickly became the fundamental protocols in use for connecting the Internet. Since its development, modifications and extensions to the protocol suite have been required to address new capabilities, constraints and technologies on the Internet. Much of the added capability has been implemented in the form of add-ons to the protocol. These additions to the protocol's capabilities have had to deal with an environment which was built without fundamental support. This has resulted in solutions which are not as efficient as they theoretically could be. The introduction of IPv6 gives the protocol designer an opportunity to build fundamental capabilities into the protocols to make future extensions, and current extensions more robust and efficient.

IPv6, or *IPng* (*IP next generation*) as it is sometimes referred to, is a totally revamped version of IP which must be interoperable with the old version while adding to the capabilities and reducing overhead. The fundamental driving force behind the development of IPv6 is the quick depletion of IPv4 addresses that is occurring in the Internet. The lifespan of the IPv4 address space has been increased through methods such as Classless Inter Domain Routing [Int87], but the Internet should still run out of IPv4 addresses shortly into the next century. One of the key features of IPv6 is its larger address space which has been increased from 32 bits to 128 bits. This increase in address space has been a major influence in the development of IPv6.

The current version of IPv6 is based upon several early proposals which were developed in the years preceding 1992. Through evaluations and discussions, these proposals were combined to form the IPng proposed specification. IPng was recommended at the Toronto IETF meeting on July 25, 1994 and is documented in RFC1752. It was also documented in "The recommendation for the IP next generation protocol" by the Internet Engineering Steering Group (IESG) on Nov. 17, 1994. This officially marks the beginning of IETF's development of proposed standards for the individual aspects of IPv6.

10.2 IPv6 Addressing

The address space in IPv4 is restricted to 32 bits. This address space is divided into a network address and a host address which together uniquely identify each host on the Internet. Extensions to this addressing have resulted in broadcast addresses and multicast addresses being allocated out of this same address space. The result is a further restricted address space with some partitioning of addresses according to functionality and service providers.

The IPv6 address space has 128 bits. This represents an immense number of addresses, equivalent to 6.66×10^{23} addresses for each square meter of the earth's surface [HD95]. Of course, such an immense address space begs to be assigned some type of order by assigning consecutive addresses to different address classes. Introducing such a class structure to the addresses, the number of host addresses is reduced, but it is still immense (between $1.564X10^3$ and $3.911X10^{18}$ addresses for each square meter by some estimates [Hui94]).

It should be noted that although the presence of a structure in the addressing scheme will reduce the shear number of addresses easily available, this structure is a very powerful tool built-in to the IPv6 protocol. By selecting groups of addresses to represent different types of addresses, this reduces the requirements of hosts on the Internet to maintain and manipulate information needed to impose these groups on a relatively non-structured addressing scheme such as IPv4. Embedding more structure and classification into the addressing scheme also simplifies protocols which need to use the IPv6 addresses to deliver their packets.

There are three main types of addresses in the IPv6 proposal. Unicast addresses are associated with a single interface (not host) on the Internet. These are the addresses commonly used to send a packet to a single host somewhere on the Internet. Every interface is assigned at least one unicast address for communication. *Anycast* addresses are new to IPv6. These addresses define a group of interfaces by a single address. Such an identification allows packets to be sent to any host in a specific group such that the nearest host will process the message while other hosts in the group will never receive it. Anycast addresses are actually just unicast addresses which have been assigned to multiple interfaces; therefore once an interface intercepts an anycast packet, no forwarding is done. Multicast addresses again map many interfaces onto a single address, but packets addressed to a multicast address will reach every interface in the group in a fashion which minimizes the Internet traffic generated.

The address space itself is allocated into several classes of addresses. These addresses are presented in Table 3.

In addition to its increased address space and other benefits, IPv6 also has an improved method for handling IP options. IPv6 has six different types of extension headers: Hop-by-Hop Options, Destination Options Header-1, Destination Options Header-2, Routing Header, Fragment Header, and the Authentication Header. Out of all these options, only the Hop-by-Hop option is processed by all intermediate routers along the path. In IPv4, any packet with options had to be processed at every router along the path. Because of the overhead involved with processing options, some implementations refuse to except packets with options and discard them. The IPv6 proposals are much more efficient in terms of processing power, and therefore there will be less desire for non-implementation of any special features.

10.3 Security Considerations

One of the enhancements that IPv6 has made possible is increased security. IPv4 has various security problems and lacks effective *privacy* and *authentication* mechanisms. In IPv6 these shortcomings are addressed by two integrated options for security services [Atk95c], the *IPv6 Authentication Header* [Atk95a], and the *IPv6 Encapsulating Security Header* [Atk95b].

The authentication header is a security mechanism for providing integrity, authentication and replay protection for IP datagrams between two Internet hosts. The authentication mechanism involves inserting an authentication header between the IP header and the IP data. The authentication header includes a security number calculated from all of the packet's static contents and known security keys. This provides enough security for most users because it guarantees the contents have not been tampered with in transit and that the source is the one involved with the

Table 3: The IPv6 Address Space [HD95]

Address Class	Address Prefix	Fraction of Total Space
Reserved	0000 0000	1/256
Unassigned	0000 0001	1/256
Reserved for NSAP Allocation	0000 001	1/128
Reserved for IPX Allocation	0000 010	1/128
Unassigned	0000 011	1/128
Unassigned	0000 1	1/32
Unassigned	0001	1/16
Unassigned	001	1/8
Provider-Based Unicast Address	010	1/8
Unassigned	011	1/8
Reserved for Geographic-based Unicast	100	1/8
Unassigned	101	1/8
Unassigned	110	1/8
Unassigned	1110	1/16
Unassigned	1111 0	1/32
Unassigned	1111 10	1/64
Unassigned	1111 110	1/128
Unassigned	1111 1110 0	1/512
Link Local Use Addresses	1111 1110 10	1/1024
Site Local Use Addresses	1111 1110 11	1/256
Multicast Addresses	1111 1111	1/256

secure communication. The details of security and authentication are beyond the scope of this document but can be found in [Atk95a].

The IPv6 encapsulation header provides confidentiality along with integrity and authentication. This support is provided by encapsulating parts of the IP datagram (either the entire datagram or only the upper-layer protocol data) inside an *encapsulated security payload* (*ESP*). The payload is encrypted using secure keys, and either tunnelled using a new IP header, or forwarded using a clear-text IP header. This mechanism should guard against snooping by insecure hosts along the route [Atk95b].

Both of these mechanisms have performance costs at the sender and receiver. For the encapsulation security header, there are hardware mechanisms for handling the encryption of the ESP which can speed things up considerably. The authentication header requires increased computation at the sender and receiver and may increase latency because of the authenticating done at each host for each datagram. Both of these mechanisms, however, provide a level of security far above that available using current IPv4 methods. Security is an important consideration with today's technology and the growth of Internet use. One technological advancement which has increased concern about security is the development of mobility supporting mechanism within the IP protocol suite [Atk95c].

11 Mobility Support in IPv6

IPv6 allows new technologies to be integrated into the protocols right from the start. One of these technologies is mobile internetworking. Since mobility requires special routing techniques and support, it is a natural prospect for consideration when developing the IPv6 protocols. Mobile IP has already been developed and is being refined for IPv4. The main protocol being considered for Internet-wide use is now the IETF Mobile IP protocol. For IPv6, the definitions for IETF Mobile IPv4 have been extended by Charles Perkins and David Johnson[PJ96]. This proposal along with the VIP proposals from Sony [TU95] are two contenders for mobility support in IPv6. Both of these options will be presented and examined in this section.

11.1 Mobile IPv6

The IETF Mobile IP for IPv4 has emerged as the Internet standard with respect to research. This proposal has been extended to fit into the infrastructure provided by IPv6. The detailed proposal for IETF mobile IP in IPv6 can be found in [PJ96], and some of its highlights are discussed here.

Mobile IP in IPv6 is made easier by the inclusion of mobile-friendly mechanisms into the basic protocol. This allows the IPv6 version of mobile IP to be simpler than that required for IPv4. The mechanisms included in IPv6 are: authentication headers, address autoconfiguration, guaranteed encapsulation support, multiple addressing capability, and a sophisticated optional routing header.

Authentication headers allow mobility in IPv6 to be secure by allowing the authentication of a host claiming to be a roaming MH. Without this security, any host could claim to be the MH and steal its packets. Also, malicious hosts could cause incorrect location updates to be sent, and the MH would loose connectivity.

Address autoconfiguration is a method by which a mobile host can acquire a globally routable address at each new point of attachment. This is a big help in hosts that are roaming and constantly need new contact interfaces. Once an MH has a co-located care-of address, it can start communicating using this address, and the routers are aware of its location.

If the encapsulation option is used for routing, the MH is guaranteed to have support. Every IPv6 host and router is required to support encapsulation. This means that an MH will always be able to find an FA on any network or subnet it wishes to attach to.

Mobile computers roaming the Internet in IPv6 will have at least two addresses for contact. There is built-in support for a host having multiple addresses in IPv6. The idea of an address uniquely identifying an individual host is gone. An IPv6 address may identify an individual interface which may coexist with other interfaces on a single host, or it may identify many interfaces which exist on many different hosts. More interfaces per host allow for the possibility of a host maintaining multiple subnet connections when it is possible to lessen or eliminate the packet loss problems associated with moving from one subnet to another.

Like IPv4, IPv6 includes an optional routing header. Routing headers allow the source of a packet to specify the routers the packet must go through in order to reach a host. This header can be used to specify a direct route to the care-of address of the MH by specifying this care-of address as the last router in the list (or the only one). This method of routing corresponds to loose source routing in IPv4; however, its specification is cleaner in IPv6, and the overhead associated with it is lower. When the host with the care-of address receives the packet, it can easily forward the packet to the MH. If the care-of address if co-located, then this forwarding is trivial [PJ96].

11.1.1 Routing and Routing Short-cuts

The basic routing in Mobile IPv6 is still the encapsulation procedure involving the HA and the FA. In IPv6, however, there are other options to encapsulation, and as well, there need not be an FA to intercept the packets at the other end if the host has an autoconfigured address. Notification about an MH's location is handled using binding update messages. Much of this terminology should be familiar, since the IPv6 protocols seem to combine attributes from many different Mobile IPv4 proposals into a single system, and the terminology comes directly from those proposals.

In IPv6, it is expected that correspondent hosts will cache the MH's location information in a mobility binding cache. In this way, it is expected that the HA will not be involved with most packet transmissions. This helps to decrease the reliance on the HA but causes extra work for the MH. When an MH moves, it is responsible for sending binding cache updates to the HA and any CHs it is aware of that have binding cache entries for the MH. In IPv6 it is required that all hosts support mobility binding caches for tracking MHs. When the HA on the home network receives the binding cache update message, it sends reachability messages out to local hosts advertising reachability to the MH [PJ96].

11.1.2 Summary

The IETF proposal appears to be a good blend of the IPv4 Mobile IP and features which were not included in that proposal due to IPv4 limitations. Some of these features existed in earlier proposals but were deemed to be impractical at the time. Other features have arisen out of new capabilities existing in IPv6. By deciding upon such a mixture of abilities, the developers of mobile IPv6 are maintaining compatibility with past protocols while allowing new features to emerge. The debate over the short-cut routing benefits and mechanisms continues, but it is clear that moving from a single failure point (the HA) to a more distributed system is better. Whether the overhead associated with updates is justified by the gains remains to be seen.

11.2 VIPonIPv6

One of the proposals being considered for mobility support in IPv6 is a version of the VIP system that was described in Section 6.2. Although it would seem natural to promote IETF Mobile IP from IPv4 to IPv6, it makes sense to at least consider some of the other ideas that were proposed in the past. One of the main arguments against some of the proposals was the need to change routers throughout the Internet. The authors of the VIP proposal argue that IPv6 represents an opportunity to move past the inefficiencies of the IETF proposal because all routers will be changed when they are changed to IPv6. It is therefore guaranteed that the required installation base will be present in an IPv6 Internetwork.

Four essential problems are identified that need to be addressed in order to support mobility. The first problem is the reassignment of network parameters when an MH moves to a new subnet. In IPv6, a process called autoconfiguration is defined which allows a host connecting to a subnet to acquire a local address and otherwise configure itself automatically by communicating directly with other IPv6 hosts on the subnet. In IPv4, specific mechanisms had to be added onto the protocols to enable this functionality.

The second problem involves TCP operation over a Mobile IP protocol. TCP is designed to work in an environment where the main cause of packet loss is congestion. To counteract congestion, TCP has a back-off mechanism which quickly slows the transmission rate of the packets. In a mobile environment, packets are lost during location changes, and can also be lost due to wireless connections. In these cases TCP backs off when it shouldn't. This phenomenon has been studied, and there is a mechanism suggested within VIP which handles this problem. It will not be discussed in this paper.

The third and fourth problems are directly applicable to the Mobile IP routing protocols themselves. Seamless mobility is the ability to recognize, and correspond with a host via the same interface regardless of its location on the Internet. Impersonation is a problem that mobility support introduces. In a non-mobile environment, a host must be on its own subnet to receive packets. For this reason, only other computers on that subnet can masquerade as the host and steal its packets. Mobility allows hosts to receive packets on subnets other than their home. This introduces the possibility of hosts away from the home subnet (and therefore hosts that are less trustworthy) being able to impersonate the MH.

Seamless mobility is handled in the same way as it was in VIP. Each MH has associated with it both a physical and a virtual address. All transactions are addressed to the virtual address (which may be the same as the physical) and are delivered using normal IP mechanisms to the physical address (which may change). Translations between these two addresses is handled at the VIP network sublayer which lies between the network and the transport layers.

Impersonation is a problem that can be addressed through authentication. IPv6 has authentication headers built in, and these can be used to address mobile security issues. VIPonV6 requires HAs and MHs to be authenticated with one another. In addition, intermediate hosts holding a cache entry for the MH needs to be authenticated with the HA before they are allowed to intercept and forward packets.

VIP uses the Hop-by-Hop option header to deliver its packets to the destination. This header forces all intermediate routers to process the packet which allows routers holding cache entries for the MHs to redirect the packet. VIP itself defines two optional headers: data and control. The data header contains nine fields, and the control header eight fields. These headers are shown in Figure 7(a) and Figure 7(b). A full description of these headers is given in [TU95].

When connecting to a new subnet, an MH transmits a control packet to its home subnet. This control packet contains registration information that is stored by a *primary address resolver* (similar to HA). This packet contains authentication information to prove the MH's identity. Forwarding routers are required to update their cache entries if the authentication data is valid.

VIPonV6 is a valid contender for a mobility support mechanism within IPv6. There are two potential problems with the protocol. One is the overhead involved with processing the option headers at all intermediate routers. It was a major advantage to have IPv6 ignore most options at intermediate routers, and this regresses the situation back to that with IPv4, however it is required for the short-cut routing ability in VIP. Another problem is with IPv4 compatibility. It is true that in a totally IPv6 environment many of the problems regarding the IPv4 version are gone or lessened substantially. The reality, however, is that IPv4 may be with us for a long time to come. In fact, it has been said that IPv4 will probably still exist on the Internet when IPv6

	Option Type	Option Data Length
Address Version		
Holding Time		
Timestamp		
Identifier (16 octets)		
Address (16 octets)		
Authentication Data (64 octets)		

(a) Mobility Control Header Option

	Option Type	Option Data Length
Source Address Version		
Holding Time		
Timestamp		
Source Identifier (16 octets)		
Authentication Data (64 octets)		
Destination Identifier (16 octets)		
Destination Address Version		

(b) Mobility Data Header Option

Figure 7: VIP Packet Headers in IPv6

becomes out-of-date. For this reason, backward compatibility is still a concern. It is still possible that VIP could work in a mixed environment, however this needs to be further investigated. In some ways VIP is a more elegant interpretation of mobility support, and perhaps it should be supported in IPv6 as an option to IETF Mobile IP, as token ring, ethernet, and wireless protocols are optional at the link-level and hardware level.

At the IETF meeting held in June 1996, it was recommended that the developers of the VIPonV6 draft work on addressing deficiencies in the proposal. Deficiencies outlined included sourcing of multicast packets by mobile hosts, firewall traversals, and the appropriateness of routing header usage [Sol96].

12 Summary and Conclusions

This paper has introduced mobility support in the Internet environment. Background information on networking, TCP/IP, and the Internet were provided as a basis for later discussion. Past protocols and current research interests have been described. Finally two options for mobility support in the next generation of IP have been discussed.

It is clear that mobility is going to be an ever-increasing factor in the Internet community. This paper provides a basis for understanding what infrastructure is being created to support this mobility. Mobile IP has been developed as an extension to IP. In order to understand Mobile IP one must first have a little understanding of the base IP protocol itself.

The Internet Protocol was developed in the 1970's as a research project investigating internetworking. The protocol was adopted as the internetworking protocol for the U.S. government defence network shortly thereafter. The address space was too small, and it was not designed for its eventual destiny, but it was working. The Internet protocol has continued to work with its limitations being addressed and fixed through modifications.

Mobility support is an extension to IP that is going to be fundamentally required in the near future. The race to design the Mobile IP protocol began in 1990 and 1991 with the development and release of at least four major protocols. Experimentation and analysis showed weaknesses in these proposals. Eventually, an IETF working group was formed and development was started on IETF Mobile IP. The IETF Mobile IP defines a complete set of mechanisms for implementing mobility support in IPv4.

The mechanisms described in the IETF Mobile IP proposal for IPv4 can be easily included in IPv6 protocols, and therefore is the basis for the IETF's proposed mobility support in IPv6. This proposal contains not only the IETF mobile IP mechanisms, but also mechanisms proposed in early Mobile IP protocols which were not advisable in an IPv4 environment. Another option for mobility support in IPv6 is the VIP protocols developed by Sony. These protocols are modelled on a more elegant view of mobile hosts on the Internet, but they may still have problems with IPv4 interoperability.

The proposals presented in this paper are both new and relatively old. Questions still remain about some protocol details, but the standards must quickly be set so that industry implementation can fully begin. Thus there is pressure to push Mobile IP through the standardization process.

The basic Mobile IP is a simple, yet effective way to provide mobility on the Internet. There are problems associated with Mobile IP, such as the security risks associated with allowing critical mobile host information to originate from a foreign network. Other problems include the potential inefficiencies and lack of robustness associated with having a single home agent on the home network forwarding all packets to the mobile host. Problems with Mobile IP have been discussed and can be addressed through extensions to the protocol.

Security concerns can be addressed by having an authentication mechanism between the home agent and the mobile host. The foreign agent and correspondent hosts must be trusted with routing the packets, as ordinary routers are trusted in the Internet today. Extensions such as short-cut routing make the security concerns more complex. This extension allows correspondent hosts to intercept packets. Therefore the routing is susceptible to these correspondent hosts receiving faulty information. Hosts masquerading as the mobile hosts can send bad location updates and other messages which can mess up MH routing from correspondent hosts. This results in a requirement for correspondent host authentication.

In general, the extra complexity and overhead associated with short-cut routing is not worthwhile unless the caching mechanisms are flexible (i.e., anyone can cache), and efficient (i.e., the network is not flooded with updates when MHs change location or other changes occur). The first problem can be solved simply by allowing anyone to cache the mobile host's current location on the Internet. Having this information is no different than having the mobile host's permanent address, which the CH must have to do any damage anyway. If a malicious host wants to trap and discard packets to the mobile host's permanent address, it can do so regardless of what security mechanisms are built into the mobile protocol. In addition, if all location update information must come from the mobile host (either directly or indirectly via the HA), and it is authenticated, then malicious hosts can not attack the system in that way. Additionally, if all MH location updates are handled by the home agent, mechanisms could be set up to allow the correspondent host to be assured of the validity of location update messages for mobile hosts which would not require off-line configuration of authentication codes. A new proposal for secure route optimization in IPv4 is due soon. The new proposal is expected to be similar to route optimization support in IPv6, and will hopefully address the concerns that have been raised about route optimization.,

The performance impact of managing multiple forwarding agents for a single mobile host is generated by the requirement to inform these hosts of location updates. If these updates are sent by the MH every time it moves, a frequently moving MH can create lots of unnecessary network traffic updating cache agents that may not even send another packet. In addition, tracking all of these routing agents requires CPU power and memory which can be scarce on portable hosts. A method in which the HA is responsible for updating location caches would be better since the HA in most cases has more resources. The idea of sending out messages to every cache agent known by the HA is potentially a waste of bandwidth. One potential method of managing conflicts is to have the correspondent hosts continue to send packets to their cached location for the mobile host. The packets should still reach the mobile host if the foreign agent is also a cache agent. The source of re-routed packets could be informed of location updates by the HA which would be informed of the update need by the foreign agent receiving the misdirected packet. A short-cut routing system using the HA as the location updater and not requiring authenticated cache agents would have adequate security, and little performance loss. The drawback is that again the HA is highly involved with location updates, and is a single source of failure.

Mobile IP and its extensions need to be studied and evaluated. The basic routing issues have been settled, however refinements like short-cuts, broadcasts, and multicasting require more study. The work so far has resulted in a high quality proposal for mobility support at the network layer. Implementations of this proposal are needed so that the impact of mobility on higher protocol layers can be studied. In particular, TCP interaction with mobility needs to be studied, and may result in optional preferences which can be set at the IP level to help TCP interpret what is transpiring at the IP layer correctly and take the appropriate action.

13 Future Work

Future work in relation to Mobile IP centres upon evaluation of the protocols, refinements to these protocols, and investigations of other protocol layers to discover the effect of Mobile IP on these layers. In particular, the following research interests are some of those being pursued:

TCP and Mobility TCP has various mechanisms designed to optimize its performance over the fixed host internetwork. A growing concern is how these mechanisms handle mobility. Research in this area focuses primarily on the behaviour of the TCP back-off mechanisms for handling congestion. This mechanism can incorrectly interpret packet loss due to mobility as congestion on the internetwork. Research includes evaluating TCP performance over wireless interfaces [JBH96], and producing specifications for new TCP protocols which handle the mobility problems [HBK95] [BB94] [CI93].

Routing Extensions It is agreed by many that the basic routing proposed for Mobile IP is inefficient and not as robust as is possible. Solutions to this problem include the ability to have more than one home agent, the ability to have multiple foreign agents, but the ability of correspondent hosts to directly forward packets to the mobile host. The protocols for handling correspondent forwarding are currently being revised and updated. The current IETF proposal has facilities built-in for multiple foreign agents, however the implications of this extension on the protocol performance, network traffic, and host overhead have not been studied. Forwarding packets from old foreign hosts to newer foreign hosts, and thus creating a chain of foreign forwarding hosts is also being currently investigated. The performance gains and losses of these extensions need to be studied, and perhaps alternatives proposed. The adaptive management scheme proposed in [RB95] shows promise, but needs to be evaluated, and could prove prohibitive because of limited resources at the MH.

Mobile Applications When mobility is offered as a viable network option, some applications will benefit from being aware of this mobility. In addition, support for handling applications on mobile computers needs to be investigated. Mobile IP and its mechanisms will affect the environment that these mobile applications will encounter. The effects of Mobile IP on certain applications will need to be studied. Many research groups are studying mobile database access, Web access, and other application program usage on mobile computers, but their research is usually based on packet radio and wireless networks. Research focused on the effects of Mobile IP on applications has been limited, and needs to be expanded if Mobile IP is going to be as popular as expected.

References

[Ami] Amigos. AMIGOS project. http://www.diku.dk/research-groups /distlab/amigos/.

[Atk94] R. Atkinson. On internet authentication. RFC1704, Network Working Group, IETF, October 1994.

[Atk95a] R. Atkinson. IP authentication header. Internet Draft, IPv6 Working Group, IETF, August 1995.

[Atk95b] R. Atkinson. IP Encapsulating Security Payload (ESP). Internet Draft, IPv6 Working Group, IETF, August 1995.

[Atk95c] R. Atkinson. IP security architecture. RFC1825, IPv6 Working Group, IETF, August 1995.

[AW96] Martin Arlitt and Carey Williamson. Web server workload characterization: The search for invariants (extended version). Available via ftp at http://www.cs.usask.ca/projects/discus/discus_reports.html, March 1996.

[Bad96] B. R. Badrinath. Dataman project. http://athos.rutgers.edu:80/dataman/, 1996.

[BB94] Ajay Bakre and B. R. Badrinath. I-TCP: Indirect TCP for mobile hosts. Technical Report DCS-TR-314, Rutgers University, Piscataway, NJ 08855, October 1994.

[BCC+94] Trevor Blackwell, Kee Chan, Koling Chang, Thomas Charuhas, James Gwertzman, Brad Karp, H. T. Kung, W. David Li, Dong Lin, Robert Morris, Robert Polansky, Diane Tang, Cliff Young, and John Zao. Secure short-cut routing for mobile IP. In *USENIX Summer 1994 Technical Conference*, Boston, MA., June 1994.

[BLCGP92] T.J. Berners-Lee, R. Cailliau, J-F. Groff, and B. Pollermann. World-Wide Web: The information universe. *Electronic Networking: Research, Applications and Policy*, 2(1):52–58, Spring 1992.

[BTB95] E. G. Britton, J. Tavs, and R. Bournas. TCP/IP: The next generation. *IBM Systems Journal*, 34(3), 1995.

[BZCS96] Mary Baker, Xinhua Zhao, Stuart Chesire, and Jonathan Stone. Supporting mobility in mosquitonet. *USENIX Technical Conference*, January 1996.

[Chi95] Vineet Chikarmane. Network support for mobile hosts in a TCP/IP internetwork. Master's thesis, Department of Computer Science, University of Saskatchewan, 1995.

[CI93] Ramon Caceres and Liviu Iftode. The effects of mobility on reliable transport protocols. Technical report, Matsushita Information Technology Laboratory, 2 Research Way, Princeton, NJ, 08540, USA, November 1993.

[CK74] V. Cerf and R. Kahn. A protocol for packet network intercommunication. *IEEE Transactions on Communications*, 22(5):637–648, May 1974.

[Com91] D. E. Comer. *Internetworking with TCP/IP*, volume 1. Prentice-Hall Inc., Englewood Cliffs, NJ, second edition, 1991.

[CWB96] Vineet Chikarmane, Carey L. Williamson, and Richard B. Bunt. Implementing Mobile IP Routing in 4.3 BSD Unix: A Case Study. Technical report, Dept. of Computer Science, University of Saskatchewan, April 1996.

[CWBM96] Vineet Chikarmane, Carey L. Williamson, Richard B. Bunt, and Wayne L. Mackrell. Multicast support for mobile hosts using mobile ip: Design issues and proposed architecture. Technical report, Dept. of Computer Science, University of Saskatchewan, October 1996.

[DC90] S. Deering and D. Cheriton. Multicast routing in datagram internetworks and extended LANs. *ACM Transactions on Computer Systems*, 8(2):85–110, May 1990.

[Dee89] S. Deering, editor. *Host Extensions for IP Multicasting*, RFC1112, Network Working Group, IETF, August 1989.

[DEF+94] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C-G. Liu, and L. Wei. An architecture for wide-area multicast routing. In *Proc. ACM SIGCOMM '94 Conference: Communications Architectures and Protocols*, pages 126–135, London, England, U.K., August 1994.

[DGL95] Abhijit Dixit, Vipul Gupta, and Ben Lancki. *Mobile-IP for Linux (ver 0.95), release notes.* Dept. of Computer Science, State University of New York, Binghamton, November 1995.

[DW88] S. Deering D. Waitzman, C. Partridge. Distance vector multicast routing protocol. RFC1075, Network Working Group, IETF, November 1988.

[DZ83] J. D. Day and H. Zimmermann. The OSI reference model. *Proc. IEEE*, 71:1334–1340, December 1983.

[Eri94] H. Eriksson. MBONE: The multicast backbone. *Communications of the ACM*, 37(8):54–60, April 1994.

[FLYV92] V. Fuller, T. Li, J. Yu, and K. Varadhan. Supernetting: An address assignment and aggregation strategy. RFC1338, Network Working Group, IETF, June 1992.

[FZ94] G. H. Forman and J. Zahorjan. The challenges of mobile computing. *IEEE Computer*, 27(4):38–47, April 1994.

[HBK95] Shrinivasan Seshan Hari Balakrishnan and Randy H. Katz. Improving reliable transport and handoff performance in cellular wireless networks. *Wireless Networks*, 1(4), December 1995.

[HD95] R. Hinden and S. Deering. IP version 6 addressing architecture. RFC1884, IPv6 Working Group, IETF, December 1995.

[HJ96] Alex Hills and David B. Johnson. A wireless data network infrastructure at carnegie mellon university. *IEEE Personal Communications*, 3(1):56–63, February 1996.

[HN87] Charles L. Hedrick and Steven E. Newton. Introduction to the internet protocols. 1987.

[Hui94] C. Huitema. The h ratio for address assignment efficiency. RFC1715, Network Working Group, IETF, November 1994.

[IDM91] J. Ioannidis, D. Duchamp, and G. Q. Maguire. IP-based protocols for mobile internetworking. In *Proc. ACM SIGCOMM '91 Conference: Communications Architectures and Protocols*, pages 235–245, Zurich, Switzerland, September 1991.

[Int87] SRI International. Network Reconstitution Protocol. Technical Report RADC-TR-87-38, Rome Air Development Center, June 1987.

[JBH96] Venkat K. Josyula, Richard B. Bunt, and Janelle J. Harms. Measurements of TCP performance over wireless connections. Technical report, Dept. of Computer Science, University of Saskatchewan, 1996.

[JM96a] David B. Johnson and David A. Maltz. *Dynamic Source Routing in Ad Hoc Wireless Networks*, chapter 5, pages 153–181. Kluwer Academic Publishers, 1996.

[JM96b] David B. Johnson and David A. Maltz. Protocols for adaptive wireless and mobile networking. *IEEE Personal Communications*, 3(1):34–42, February 1996.

[JMP94] D. B. Johnson, A. Myles, and C. Perkins. The internet mobile host protocol (IMHP), February 1994. Internet Draft.

[JP81a] editor Jon Postel. Internet Protocol: DARPA internet program protocol specification. RFC791, September 1981.

[JP81b] editor Jon Postel. Transmission Control Protocol: DARPA internet program protocol specification. RFC793, September 1981.

[JP96] David B. Johnson and Charles Perkins. Route optimization in mobile IP: Version 04. Internet Draft, Mobile IP Working Group, IETF, February 1996.

[LCT95] Y. Z. Li, K. C. Chua, and Y. C. Tay. An implementation of mobile IP within Linux kernel. http://zaphod.ee.nus.sg/mip/mip-doc.html, December 1995.

[Lia96] Jiang Ming Liang. Mobile IP at NUS. http://mip.ee.nus.sg/, August 1996.

[Mon96] G. Montenegro. Bi-directional tunneling for mobile ip. September 1996. Internet Draft, Mobile IP Working Group, IETF.

[Moy94] J. Moy. Multicast routing extensions for OSPF. *Communications of the ACM*, 37(8):61–66, August 1994.

[MP85] J. Mogul and J. Postel, editors. *Internet Standard Subnetting Procedure*, RFC950. USC/Information Sciences Institute, August 1985.

[MS93a] A. Myles and D. Skellern. Comparing four IP based mobile host protocols. *Computer Networks and ISDN Systems*, 26:349–355, 1993.

[MS93b] A. Myles and D. Skellern. Comparison of mobile host protocols for IP. *Proc. Internetworking Research and Experience*, 4:175–194, December 1993.

[Myl92] A. Myles. IP options. mobile IP mailing list, 29, July 1992.

[Myl93] October 1993. Internet Draft.

[Myl94] Andrew Myles. Andrew myles - mobility archive. http://www.mpce.mq.edu.au/ andrewm/mobile.html, 1994.

[NSF95] NSFNET statistics. Nsfnet ftp server. ftp://nic.merit.edu/statistics/nsfnet, 1995.

[Par94] Craig Partridge. *Gigabit Networking.* Addison-Wesley, 1994.

[Per95a] Charles Perkins. IP encapsulation within IP: Version 01. Internet Draft, Mobile IP Working Group, IETF, October 1995.

[Per95b] Charles Perkins. Minimal encapsulation within IP: Version 01. Internet Draft, Mobile IP Working Group, IETF, October 1995.

[Per96] C. Perkins. IP mobility support: Version 16. Internet Draft, Mobile IP Working Group, IETF, April 1996.

[PJ96] Charles Perkins and David B. Johnson. Mobility support in IPv6: Version 00. Internet Draft, IPv6 Working Group, IETF, January 1996.

[PR92] C. Perkins and Y. Rekhter. Short-cut routing for mobile hosts, July 1992. Internet Draft, T.J. Watson Research Center, IBM.

[RB95] Subhashini Rajagopalan and B. R. Badrinath. An adaptive location management strategy for mobile IP. In *MOBICOM*, 1995.

[Rei94] Andy Reinhardt. The Network with Smarts, October 1994. Byte Magazine.

[Sol95] J. Soloman. Applicability statement for IP mobility support: Version 01. Internet Draft, Mobile IP Working Group, IETF, December 1995.

[Sol96] James Soloman. Mobile IP minutes. mobile IP mailing list, July 1996.

[SP80] C. Sunshine and J. Postel, editors. *Addressing Mobile Hosts in the ARPA Internet Environment*, IEN 135, March 1980.

[Tan88] A. S. Tanenbaum. *Computer Networks*. Prentice-Hall, Inc., Englewood Cliffs, N.J., second edition, 1988.

[TU95] F. Teraoka and Keisuke Uehara. Mobility support in IPv6 based on the VIP mechanism. In *Network Technology: INET'95*, Honolulu Hawaii, June 1995.

[TUSM94] F. Teraoka, Keisuke Uehara, Hideki Sunahara, and J. Murai. VIP: A protocol providing host mobility. *Communications of ACM*, 37(8):67–75, 1994.

[TYT91] F. Teraoka, Y. Yokote, and M. Tokoro. A network architecture providing host migration transparency. In *Proc. ACM SIGCOMM 91*, pages 209–220, Zurich, Switzerland, September 1991.