Addison-Wesley Wireless Communications Series

Mobile IP

Design Principles and Practices

Charles E. Perkins

Mobile IP

The Addison-Wesley Wireless Communications Series
Andrew J. Viterbi, Consulting Editor

CDMA: Principles of Spread Spectrum Communication
Andrew J. Viterbi

Wireless Personal Communications Systems
David J. Goodman

Mobile IP: Design Principles and Practices
Charles E. Perkins

Wireless Multimedia Communications: Networking Video, Voice, and Data
Ellen Kayata Wesel

Please see our web site (http://www.awl.com/cseng/wirelessseries)
for more information on these titles.

Mobile IP

Design Principles and Practices

Charles E. Perkins

ADDISON–WESLEY
An Imprint of Addison Wesley Longman, Inc.
Reading, Massachusetts • Harlow, England • Menlo Park, California
Berkeley, California • Don Mills, Ontario • Sydney
Bonn • Amsterdam • Tokyo • Mexico City

The publisher offers discounts on this book when ordered in quantity for special sales. For more information please contact:

Corporate & Professional Publishing Group
Addison-Wesley Publishing Company
One Jacob Way
Reading, Massachusetts 01867

Library of Congress Cataloging-in-Publication Data

Perkins, Charles E.
Mobile IP : design principles and practices / Charles E. Perkins.
p. cm.
Includes bibliographical references and index.
ISBN 0-201-63469-4
1. Mobile computing. 2. TCP/IP (Computer network protocol)
I. Title.
QA76.P425 1997
004.6'2—dc21 97-35781
CIP

ISBN 0-201-63469-4
Text printed on acid-free paper
4 5 6 7 8 9—MA—01009998
Third printing, May 1998

CONTENTS

LIST OF FIGURES

Chapter Seven

Chapter Eight

Chapter Nine

PREFACE

chnological advances in recent years have radically altered the nature of ing for most computer users. The first is mobility. Laptop computers now at the fastest growing segment of the computer market. Most observers hat laptop computers, palmtop computers, networked personal digital as- and other such mobile computers will eventually represent the majority tations connected to the Internet. The advantage of mobile computing is rs may access all their applications from any location, whether they are in building or a different state. The second advance is the widespread use ternet for communication, file transfer, and World Wide Web connectivity. ok describes how to make a mobile computer user a citizen of the Internet v to access everything the information superhighway has to offer.

goal of this book is to provide you with an introduction to the design and entation of Internet protocols that are useful for maintaining network con- even while moving from place to place. We look at several protocols g Mobile IP, route optimization, IP version 6, the Dynamic Host Configu- rotocol, encapsulation, source routing, and some related topics still under ment.

se full advantage of the information in this book, you should be familiar with ternet protocols such as the Transmission Control Protocol (TCP)/Internet (IP). Rich Stevens' book *TCP/IP Illustrated, Volume 1: The Protocols* (Stevens d Douglas Comer's *Internetworking with TCP/IP* (Comer 1991) both provide t introductions to TCP/IP. As a developer of hardware and software prod- the Internet, you should have these books on your shelves.

e time you finish *Mobile IP: Design Principles and Practices*, you will be able ment Mobile IP, and will have a clear understanding of the system impact lity. You will also understand the relevant protocols, and the traps and hat you are likely to encounter along the way.

ou read this book you will notice many italicized terms, some of which nventional meanings that may be different than one's first impression (for , *foreign agent*). These terms are defined in the Glossary: Please check ns there, and be sure that you understand a term's meaning before moving e text.

ACKNOWLEDGMENTS

Much of the material in this book has been adapted from protocol specifications text. In particular, Chapters 3 and 4, which detail the base IETF Mobile IP specification (RFC 2002 (Perkins 1996b)), are similar to the relevant part of that protocol specification. As editor of the Mobile IP drafts, I gratefully accepted text for inclusion from a number of people, all of whom have contributed to the success of this effort. Dave Johnson and Jim Solomon are undoubtedly the two biggest contributors.

Chapter 5, which deals with encapsulation, was largely drawn from RFC 2003 (Perkins 1996a) and RFC 2004 (Perkins 1996c), with material added from the base specification to describe the ways that Mobile IP routes various types of datagrams. Thanks to Tony Li for his generous permission to allow me to use the General Record Encapsulation documentation (Hanks et al. 1994a, Hanks et al. 1994b) to prepare the relevant text in Chapter 5. Parts of Sections 5.2 and 5.6 were taken from portions (authored by Bill Simpson) of earlier versions of the Mobile IP Internet draft (Perkins 1995). The original text for Section 4.9 was contributed by Bob Smart. Good ideas have also been included from RFC 1853 (Simpson 1995).

Thanks also to Anders Klemets for finding mistakes and suggesting improvements. Again, Dave Johnson contributed a great deal of time grooming the drafts, finding mistakes, improving consistency, and making many other improvements to the numerous Internet drafts.

Chapter 8 expands on a paper that I coauthored with Dave Johnson and presented at Mobicom '96 (Perkins and Johnson 1996). Dave also collaborated on Route Optimization (Chapter 6). The multicast preference extension described in Chapter 7 was originally presented as an Internet draft (Bhattacharya, Patel, and Perkins 1996), coauthored with Baiju Patel and Partha Bhattacharya from IBM.

Thanks to Steve Deering (Xerox PARC), along with Dan Duchamp, Chip Maguire, and John Ioannidis (JI) (Columbia), for forming the working group, chairing it, and putting so much effort into its early development.

Thanks also to Kannan Alaggapan (DEC), Greg Minshall (Novell), and Tony Li (Cisco) for their contributions to the Mobile IP effort, as well as for their many useful comments. Thanks to Greg Minshall, Phil Karn (Qualcomm), and Frank Kastenholz (FTP Software) for their generous support in hosting interim working group meetings.

Special thanks to my friend Andrew Myles, who not only battled with me for long hours via telephone and electronic mail about the right ways to do things, but delighted my children and enlivened our wireless systems lab with his insight, hard work, and working code. Thanks to Pravin Bhagwat (IBM) and Felix Wu (IBM) for their diligent work in helping to implement our early Mobile IP specifications. Thanks to Tangirala Jagannathan (IBM) for persisting through the ever-more-detailed implementation requirements imposed by later drafts, for having the willpower to become expert in the vagaries of our previous implementations, and for handling the battle against the Dynamic Host Configuration Protocol so well. Thanks to Hui Lei (Columbia) for his efforts to make Mobile IP compatible with ad hoc network protocols, and to Kavitha Devara for her efforts in creating last ditch demos. I'm happy that developing and implementing the Mobile IP specifications has enabled me to enjoy the added bonus of counting these people as my friends.

CHAPTER ONE

Introduction

Computing in the 1990s is being transformed by an inexorable march toward greater user convenience, greater processing power, more storage, and better display technologies. From humble beginnings with small diskette-based systems with only a few kilobytes of memory, the personal computer has grown to become a truly transportable device with dozens of megabytes of main memory, gigabytes of disk storage, orders of magnitude more processing power, and beautiful color displays that seemed unimaginable in the early 1980s. Laptop computers should no longer be considered the poor cousins of workstations or even mainframes, but should be thought of instead as another choice in a wide spectrum of available computer resources.

Just as there has been an unstoppable trend toward having additional computing power at one's fingertips, the world of networked computing has similarly advanced at an amazing pace, approximately doubling in connectivity and reach every year. In other words, the number of computer users connected to the network next year is likely to exceed the total number of network-connected people in each previous year added together. This rate of growth is causing revolutionary changes in network technology development and indeed has necessitated social, business, and legal advances for integrating the technology into everyday life.

This book furthers such revolutionary changes by demonstrating new ways to view the connections between mobile devices and the ever-growing worldwide network of computing resources. As people move from place to place with their laptop, keeping connected to the network can become a challenging and sometimes frustrating and/or expensive proposition. The goal is that with the widespread deployment of the mobile networking technologies described here, automatic communications with globally interconnected computing resources will be considered as natural for people on the move as it is for people sitting at a high-performance workstation in their office. In the near future, communicating via laptop should be as natural as using a telephone.

The day will arrive, hastened by Mobile IP, when no person will ever feel "lost" or out of touch. Indeed, with sufficient connectivity (and a network of trusted friends and family), one could issue an alarm at the first sign of danger. Even today, global positioning system (*GPS*) systems are used to assist in quickly

determining personal location, and knowing one's current location is a first step toward getting help sent to where it is needed. Combining GPS data with access to Internet data relevant to the coordinates of the mobile computer user means the possibility of more effective action and a greater sense of personal security. Moreover, one can obtain information from Internet data sources about events that have recently affected an area. Similarly, the likelihood of preventative action may have a revolutionary effect on the incidence of violent crime. Mobile IP can also further enhance today's pagers and cellular telephones by allowing natural access to Internet data.

In this introductory chapter, after a short overview of the relevant, existing network protocols, the essential problem solved by Mobile IP is described. The two conflicting requirements for a changeable network address (for routability) and a stable network address (for identification purposes by transport protocols, notably TCP), are reconciled by introducing a level of indirection in the network that then introduces a need to maintain associations between the two network addresses involved. After this discussion, an abstract model of mobile networking is presented that shows the nature of the mobile networking problem and describes some possible solutions. The functions needed for managing the addresses are then described, and particular instances are identified by analyzing some sample designs.

Mobile networking fits in the larger context demanded by the need for total solutions to the problem of nomadic computing and the system support envisioned for solving the needs of nomadic users. Nomadic computer users bring new requirements that affect every layer of the network protocol stack. Some new application-level requirements are described, including

- Dynamic resource and service discovery
- Coping with dynamically changeable *link* conditions
- Profile management
- Environment management
- Proxy services

As this book uses a great number of new terms, which have meanings that are not always obvious, a glossary is provided at the end of the book to define unfamiliar terms.

1.1 Laptop Computing

Although computers can be embedded in a wide variety of mobile systems, the first and most important mobile computer system of interest is undoubtedly the laptop computer, which is rapidly becoming indispensable for the business trav-

eler. A typical laptop system can be equipped with a high-resolution color display, multiple gigabytes of disk space, high-fidelity audio output supported by a digital signal processor, a pointing device, high-speed network connections, a battery with enough electrical storage to last an entire business day, and a wireless communications adapter.

There are many kinds of communications adapters that allow convenient access to modern computer networks, and laptop computers typically come equipped with networking software to transmit and receive data over those networks. For access to the Internet, laptops must have the correct protocols, namely TCP/IP and the various auxiliary protocols associated with electronic mail, Web browsing, and other Internet functions. With this in mind, and because the basic equipment in a laptop is as capable as a desktop computer, the current goal is for laptop computers to operate TCP/IP as easily as desktop computers. The fact that this is not yet a reality is the result of insufficiencies in Internet protocols, not the result of inadequate computing power in the laptop.

1.2 Wireless Technologies

With wireless communications systems (and battery-powered operation) laptop computers can be completely tetherless and still have full connectivity to the Internet. New technologies are available that boast faster transmission speeds approaching those of the wired networks of only a few years ago. Wireless telephone communications provide almost complete coverage of most populated areas within the United States and Europe. Of particular interest to readers of this book are the local area network (*LAN*) attachment devices, which typically use infrared light or radio frequency signals to establish links to a wired LAN. Cellular telephone technology (which is also a radio frequency technology) is also of interest to mobile computer users, but cellular phone users rely on the telephone company to maintain connectivity and usually pay a substantial premium for that service. In contrast, radio or infrared LAN attachments are typically made without charge, as long as the LAN administrator is willing to accommodate the user's wireless link.

Here is an example of how a laptop computer might be used with an infrared communications adapter. Suppose that an installation has infrared access points installed in a user's office, in the hallways, and in the conference rooms. When the appointed time comes for an important meeting and the user has just finished completing the presentation materials for the meeting, the user can safely carry the laptop computer to the conference room without shutting down and then restoring the communication links in the new room.

To move within a building, operation at the network layer means that Mobile IP eliminates any concern about which network of the many interconnected networks is closest to the user's current connection point to the building infrastructure. Mo-

bile IP also does not depend on the physical nature of the connection between the laptop computer and the rest of the Internet. That is, it does not matter whether the computer is connected via radio LAN, infrared, wireless telephone, or indeed whether the computer is hooked up directly to an Ethernet or token ring network. Physical-layer independence is very powerful in practice. Once the network-layer protocol can accommodate the mobile computer, each new wireless or network adapter that becomes available may be used for mobile computing.

Indeed, a kind of multimodal operation is possible whereby all the software on the laptop computer can maintain connections to the Internet even though the user has changed the physical medium by which the connection is made. For instance, a user may wish to use infrared or even Ethernet links while inside a building, may switch to radio LAN connections when leaving the premises, and may use a cellular telephone to maintain connectivity when out of range of all enterprise base stations, as demonstrated by experimental programs carried out at the University of California at Berkeley (Katz 1994). Preservation of connectivity truly deserves the name *seamless roaming*.

1.3 Information Superhighway

Much has been written recently about the emergence of the Internet as the long-sought-after information superhighway. From its humble beginnings as the Arpanet of the 1970s (Cerf 1978), populated by a few dozen huge computers of the day, the Internet has been doubling in size to become a democratic and very noisy harbinger of the future of world communications. The information resources available on the Internet are as vast and varied as humanly imaginable. This is demonstrably true, because as soon as someone imagines a new computer resource that could be available on the network, it seems that resource soon emerges.

For years, communication via the Internet seemed possible only for computer specialists. Electronic mail, file transfers, and even on-line multiplayer games were almost unknown to most nontechnical people. However, the emergence of the World Wide Web and tools such as Netscape, which make the Internet accessible to everyone, have introduced another stage of tremendous expansion.

Recent traffic analyses of Internet packet flows show that a substantial majority (over seventy-five percent) of all Internet traffic is indeed caused by requests from Web browsers. Moreover, most of the Web traffic consists of image transfers. This trend is bound to continue and probably will increase in the near future. In fact, once the necessary routing protocol details are worked out for improving the delivery of video clips, the denizens of the Internet (*netizens*) will likely find ways to combine video, audio, and text into ever-more dazzling (and ever-more bandwidth-consuming) amalgamations of network-retrievable information. Judging from the continuing emergence of new magazines and television shows, there is no limit

to humanity's appetite for new visual (and cerebral) stimulation. As technology improves, prices for interconnection drop, and available storage for network information continues to grow, the Internet's role in satisfying these visual needs will continue to expand.

The Internet is also playing an ever-increasing role in the dissemination of technical reports, mail, design documents, and many other aspects of professional communications including videoconferencing. Indeed the whole nature of technical publication is changing rapidly. Reports that formerly had to wait many months before appearing in refereed journals are now distributed worldwide as preprints to anyone who may find them of interest. Professional computer engineers were among the first to equip themselves with the kind of powerful and portable laptop computer that maintains access to the Internet. Access to computer resources is a powerful motivator for improving the network connectivity for laptop computers. Now more than ever it is easy to find information on almost any subject, but one needs to learn which search engine to use, which keywords to filter, and which buttons to click.

1.4 Mobility versus Portability

This book distinguishes between two similar terms—*mobility* and *portability*. Up until now most mobile computer users have had to be satisfied with portable operation. In other words, the computer can be operated at any one of a set of points of attachment, but not during the time that the computer changes its point of attachment. If the computer is moved from one place to another, then its network connections have to be shut down and reinitialized at the new point of attachment to the network. Future mobile users will not be satisfied with this mode of operation, especially if they know that the network *could* support undisturbed connectivity between application and resource.

This book describes protocols that allow truly mobile operation, so that the laptop can remain in almost continuous contact with the network resources needed by its applications. Using these protocols, neither the system nor any of the applications running on the system need to be reinitialized or restarted, even when network connectivity is frequently broken and reestablished at new points of attachment.

Considerable effort has been put into expanding the sphere of applicability of certain existing protocols such as Point-to-Point Protocol (*PPP*) (Simpson 1994), Dynamic Host Configuration Protocol (*DHCP*) (Alexander and Droms 1997), and Domain Name System (*DNS*) (Mockapetris 1987a, Mockapetris 1987b) to support the portable mode of operation for mobile computers. This book shows that by solving the mobility problem at the network protocol layer, solutions requiring other complex protocols (such as DHCP, and extensions requiring modifications to critical enterprise subsystems like DNS) can be largely superseded by less expensive and more general technology.

It's worthwhile to point out that nomadic users of today's Internet are often satisfied with portable computing. For such users all that is needed is a temporary connection to the Internet that is broken when the time comes to move to a new place. For them maintaining connections doesn't matter much, because the connections are short lived. Moreover, often for Web-based information retrieval, the network address of the recipient does not matter. In these cases, the assumptions made about the IP network addresses being closely related to identity are not very strong, and thus Mobile IP does not provide much benefit.

Another factor that promotes solutions that minimize connection lifetime is expense. For instance, a nomadic connection maintained over a telephone link from an airplane costs well over $1 per minute. This is a great disincentive to keeping idle logins on remote computers. In an office setting no one would think twice about the expense, and workers can experience productivity gains as a result of having the additional network resources available.

It is my strong belief that as wireless computing becomes more prevalent and less expensive, and especially as wireless *cells* shrink in size to promote frequency reuse and greater cumulated bandwidth, Mobile IP will be viewed increasingly as a necessity, and the concessions made to today's realities of portable computing will be viewed more as bugs rather than necessities.

1.5 Quick Overview of IP and Routing

The Internet is largely built using software that relies, unsurprisingly, on the Internet Protocol suite, and specifically on IP (Postel 1981b). It is assumed in this book that you are somewhat familiar with IP and understand in some detail the *catenet model* (Cerf 1978) it provides, by which routers forward datagrams from one network to another by selecting the next hop that the datagram must traverse (Steenstrup 1995). For each datagram, each router in the Internet determines the next hop by finding the entry in its routing table that best matches the destination IP address of the datagram.

1.5.1 IP Addresses

The purpose of routing protocols within the Internet is to allow routers to exchange information about the networks they are connecting. As the routing information flows across the Internet, each router will eventually learn enough to send any datagram along the correct route to its destination. *Nodes* that are not routers typically accomplish this objective simply by sending all of their outgoing datagrams to a default router.

Routers have a difficult task because they have to decide how to forward each packet they receive, a decision that involves selecting from several outgoing network interfaces to forward a packet. Even so, most routers only need to keep track of a

small proportion of the total number of routes within the Internet. Routers can try to find a route (that is, a network interface) appropriate for delivering a datagram, and if their attempt does not meet with success they can send the datagram along to another default router for further handling. In this way, datagrams proceed until (as usual) they either arrive at the correct network in few a local hops or they have to go out over the national and international routers (the backbone routers), which have full (if aggregated) knowledge of all high-level routes within the Internet. As one might expect, there is a great deal of interest in finding ways to reduce the number of routes that each router, even on the backbone, needs to maintain.

IP network address allocation and administration have historically assumed that there is a close relationship between a computer's IP address and its physical location. This proceeds naturally from the assumption that a network is easily modeled by a wire (say, an Ethernet cable), and thus to a great extent can be localized. This model works equally well even if multiple cables are connected to a bigger network (say, by repeaters or bridges) (Perlman 1994). As far as the routers are concerned, the mass of cables hooked together by such devices still operates as if it were a single cable and is considered by IP to be a single network. That network is addressed by a single IP prefix (Stevens 1994), and all computers hooked together along that network are assigned addresses that use that same prefix. Among other things, this implies that any two computers connected to that network can communicate directly, without using the services of any router. Computers not on that network will have a different prefix, which can be used to locate the network to which they are attached.

Thus, IP addresses (Figure 1.1) have two parts:

1. The *routing prefix* (often determined by the *netmask*) defines the network on which the address resides.
2. The host number fits in the least significant remaining bits of the IP address following the routing prefix bits.

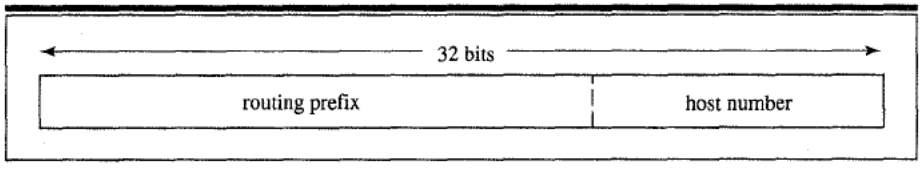

Figure 1.1 IP address structure.

1.5.2 Routing

Effectively all IP addresses are split into network prefixes and host numbers. The Internet is far too big to use flat addressing, which is when each host has its own entry in every Internet router because of the difficulty of handling updates for each

destination. Instead, router entries refer to much larger sets of hosts, namely those that are located together on a subnet. Thus, the routers use a kind of topological addressing and make the assumption that hosts with common routing prefixes can share a common route.

In the past, the routing prefix could be either class A, B, or C. The class was determined by the number of bits in the prefix. One might say that the netmask was implicitly encoded in the high-order bits of the IP address. More recently, to preserve as best as possible the remaining IP address space, routing prefixes have been assigned according to the architecture prescribed in classless interdomain routing (*CIDR*) (Rekhter and Li 1993, Fuller et al. 1993). With CIDR, the netmask is explicitly given separately from the IP address, which allows previous class A and B networks to be carved up into a much larger number of smaller networks. Furthermore, the smaller networks are usually aggregated so that fewer router advertisements are needed overall at the highest levels of the routing infrastructure.

From the point of view of routing, the problem with mobility is that mobile computers move from one IP subnet to another, but have the wrong subnet prefix for the destination subnet. For instance in Figure 1.2, the mobile computer from subnet 132.4.16 is shown attaching to a subnet with routing prefix 128.8.128. This is going to cause trouble because no datagrams for network 132.4.16 will arrive on 128.8.128.

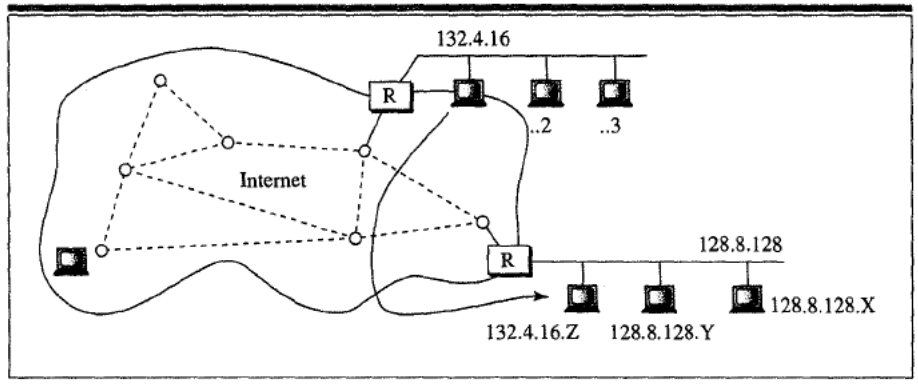

Figure 1.2 IP subnet model versus mobility.

1.5.3 Source Routing

Even though routers maintain practically all the necessary routing information used within the Internet, there are cases in which nonrouter hosts must specify certain routing information, including

- Point-to-point connections
- Multihomed hosts
- Source routing

In this book, the first two routing variations are not considered in detail. However, *source routing* has had a large and continuing effect on the development of mobile networking protocols, and is briefly described in this section.

In general, source routing means the insertion of routing information into a datagram by the node that originates the datagram (the source node). This routing information can be inserted for a variety of reasons, such as

- Policy routing
- Enabling new routes that are not otherwise advertised
- Debugging

Policy routing is a general term that means selecting a route different from the usual route to suggest or enforce a desired policy for the traversal of the datagram across the Internet. Debugging was the original motivation for the inclusion of source route options. The second use of source routes is the one that has attracted the attention of the designers of mobile networking protocols.

In IP, source routes are specified in the IP options compartment of the IP header. There are two kinds of source routes available: strict source routes and loose source routes.

Strict source routes specify every intermediate routing point that a datagram must visit. If the datagram ever arrives at an intermediate routing point that is not directly adjacent to the next hop in the list of intermediate points in a strict source route, the datagram cannot be forwarded. In that case, an Internet Control Message Protocol (*ICMP*) (Postel 1981a) error (parameter problem message) is returned to the sender and points to the node in the list after which routing was prohibited.

Loose source routing (*LSR*), on the other hand, does not prohibit the further delivery of a datagram when the next hop in the source route list is not adjacent to an intermediate hop. Thus, LSR is more flexible for normal use in routing Internet datagrams. In short, the idea that occurred to several researchers in mobile networking was to define and use a current IP address associated with a *mobile node* as an intermediate hop in a loose source route accompanying all datagrams destined for that mobile node (Perkins and Bhagwat 1993, Johnson 1994).

1.6 TCP Connections

A network application often has to identify the communication endpoints that are receiving data by way of some connection over the network. For instance, if an application running on a network client needs to send a file to its remote server,

the protocols invoked by the application need to format the data according to specifications so that the protocol processing on the remote node can make sense of the data and digest it at a convenient rate. Every data transfer between network endpoints has to be tightly controlled by conformance to interoperable network protocols such as TCP (Postel 1981c), the User Datagram Protocol (*UDP*) (Postel 1980), or the Real-time Transport Protocol (*RTP*) (Schulzrinne et al. 1996).

This is usually done by maintaining the IP address of both endpoints as part of a *protocol control block*, which stores all of the information needed for the higher level network protocols to manage the connection between the endpoints. TCP makes available a number of *ports* to network endpoints and uses the port numbers along with the IP addresses of the endpoints to identify its protocol control blocks. Consequently, for transport protocols such as TCP, the IP addresses of the network nodes serve to identify the endpoints of the communications channels used for the data transfers, as illustrated in Figure 1.3. Figure 1.3 also shows that a routing path between the two endpoints, SH and MH, and that alternate paths may be possible.

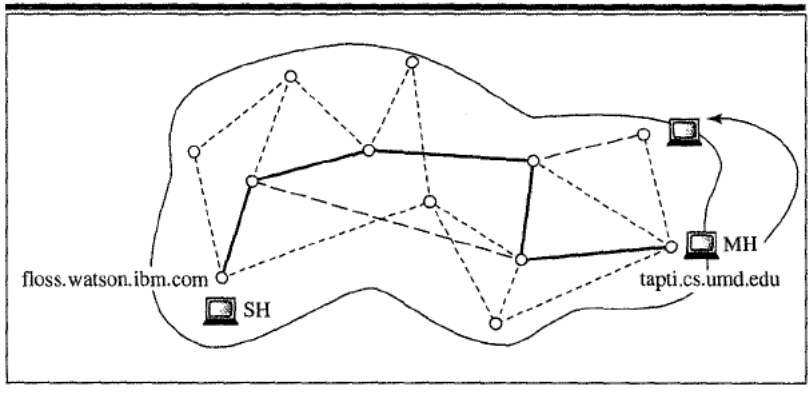

Figure 1.3 Connections between Internet computers.

1.7 Two-level Addressing

As shown in the previous sections, applications use IP addresses to identify routes by which datagrams may be exchanged between two network nodes, namely the nodes performing the actions needed for the application. On the other hand, the IP address used by the applications is also used to identify the endpoints themselves. This dual use of the IP address by the application endpoints causes problems when trying to use the application while changing one's point of attachment to the Internet. Clearly, applications need an unchanging way to identify the communication endpoints, but just as clearly the routes between the endpoints must change as they move from place to place within the Internet.

In Figure 1.4, the mobile node named FOO has moved from subnet 132.4.16 to another subnet, 128.8.128. As suggested by Figure 1.4, Mobile IP solves this quandary by maintaining two addresses—one for each of the dual uses of the IP address. Subsequent chapters detail the various mechanisms for acquiring and associating the two addresses; for now, however, the important point is that one IP address is available for *locating* the mobile computer and another is available for *identifying* a communication endpoint on the mobile computer.

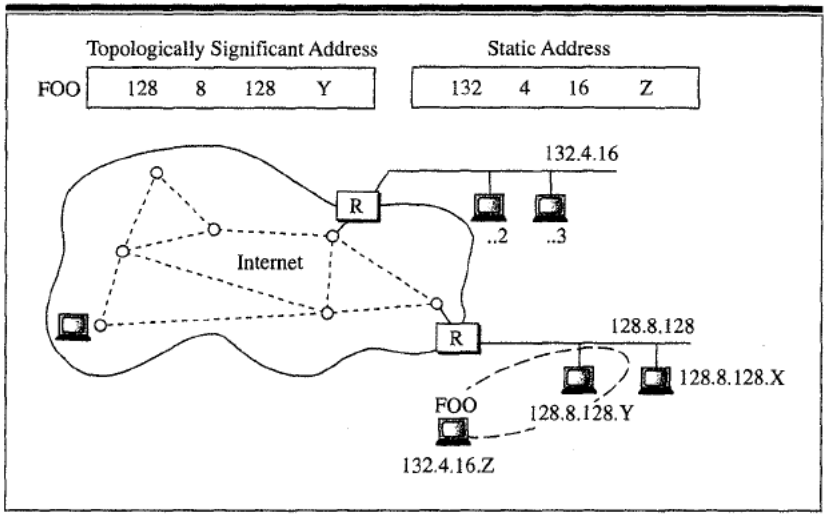

Figure 1.4 Two-tier IP addressing.

1.8 Abstract Mobility Management Model

Mobile IP is described in this book, as well as other related protocols that have been proposed by identifying the necessary functions for managing the IP addresses used for locating and identifying the mobile node. There have been quite a number of proposals (Ionnidis and Maguire 1993, Perkins and Myles, 1994, Perkins, Myles, and Johnson 1994, Teraoka and Tokoro 1993, Wada et al. 1993, among others), and it is possible to identify the necessary functions by studying the elements that these proposals have in common. Consult the article by Myles and Skellern (1993) for a detailed analysis of several of these proposals. For the purposes of this book, it is enough to describe the abstract functions and how they are represented by the functions provided by Mobile IP.

Since there are typically two IP addresses associated with each mobile node, there is a need for one or more directories to store the associations. The directory should be indexed by the IP address used to identify the mobile node to the Internet at large. Each entry should contain the associated IP address, which can

be used to locate the mobile node. This associated address is known as the *care-of address*.

As mentioned, Internet nodes typically use the IP address of a destination node when searching for connections to that destination. The IP address of a remote endpoint also serves to identify the endpoint and is used by IP as the destination address in the IP header. The IP header precedes the higher level protocol headers and payload. The destination address carries with it the indication of a particular network (that is, uniquely specifies a particular network's prefix), and the destination address is typically unchanged during transit (disregarding source routes). Therefore, the network toward which a datagram is routed usually directly depends on the identity of the desired remote endpoint, as known to the source of the datagram.

For mobile computers this destination network is known as the *home network*. Since the mobile node appears to the rest of the Internet as if it were actually located on the home network, we call its IP address the *home address* of the mobile node. If the source of the packet is an Internet node with no special modification for mobility support, it will be unaware whether anything special happens when the datagram arrives at the home network. Indeed, if the mobile node is located at the home network, nothing special needs to happen. The datagram will be delivered to its home address and thus to the intended recipient, namely the mobile node.

However, if the mobile node is not attached to its home network, then the datagram somehow needs to follow the mobile node to its care-of address. Since the datagram can only follow the mobile node by utilizing the existing Internet infrastructure for the intervening hops, it seems clear that the addressing of the datagram needs to be changed before the datagram is able to follow the mobile node off the home network.

Changing the address of the datagram for further routing is known as *readdressing*. Readdressing a datagram transforms its original destination IP address (the home address, which identifies the mobile node) into a different destination IP address (namely, the care-of address, which locates the mobile node).

The other abstract function needed for supporting mobility is just the inverse of the readdressing function. If one agent applies an address translation function to a datagram destined for a mobile node, it seems prudent (at least in the abstract) to provide for the possibility of the inverse function so that the original datagram can be presented to the mobile node. The inverse operation is required if the higher level protocols in the mobile node and the nodes with which it corresponds are to operate in a symmetrical manner. Otherwise, the mobile node's home address identifying the higher level protocol connection status control blocks would not be available in the same way as expected by the node that originally sent

the datagram. Higher level protocols do operate in this symmetric fashion, and any reasonable Mobile IP architecture must be built with the intention of reducing or eliminating any modifications to existing higher level protocols.

Stated simply, the inverse readdressing function transforms the datagram so that the care-of address (having fulfilled its role in life) is replaced by the home address (used originally by the source node as the destination IP address). It is not necessary, of course, that these individual abstract functions be performed by physically distinct Internet nodes. The exact ways in which the functions are located in various nodes and networks distinguish the various approaches.

To summarize, the following abstract functions are needed to support mobility:

- Readdressing at the home network
- Associating (in the location directory) the home address and the care-of address of the mobile node and maintaining up-to-date values for the association
- Delivering the datagram to the care-of address
- Inverting the readdressing operation once the datagram arrives at the care-of address

These functions are depicted in Figure 1.5. As shown, the readdressing function f consults the location directory (LD) to retrieve the care-of address of the mobile node before performing the readdressing operation and attempting further delivery of the datagram. When the datagram arrives at the care-of address, the inverse readdressing function g is applied and the recovered datagram is delivered to the mobile node for processing by its higher level protocols. Figure 1.5 assumes that the inverse readdressing does not need to consult another directory. This is typically true because the readdressed datagrams contain sufficient information to allow the inverse operation to proceed.

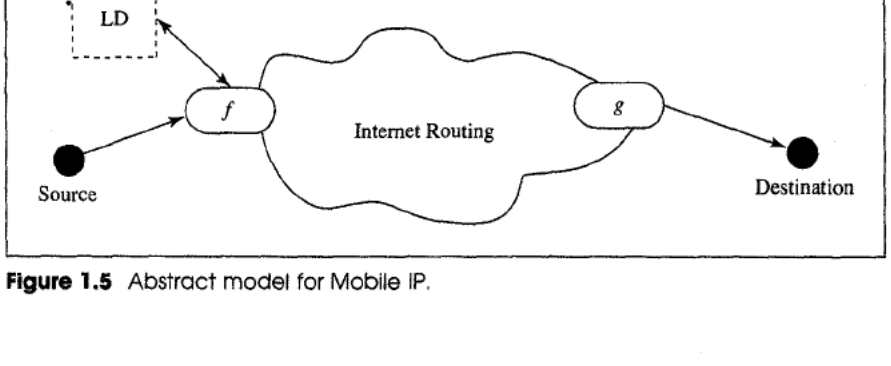

Figure 1.5 Abstract model for Mobile IP.

1.9 Remote Redirection

A common feature of all instances of the abstract architecture is the need to update the LD when the mobile node moves to a new attachment point within the Internet. The update message sent to the LD has the effect of directing traffic from the home network to the mobile node's new location. This redirection operation is known generically as a *remote redirection*, and it introduces stringent security requirements for the realistic deployment of any workable protocol. Security (in particular, authentication) is required so that the LD manager can be assured that the mobile node itself has authorized the delivery of all location update information for the LD. If bogus, counterfeit, or otherwise malicious location updates were accepted, the mobile node could be cut off from future communications with the Internet. The uncontrolled propagation of remote redirects must be avoided to eliminate such problems (Bellovin 1989, Voydock 1983).

Unfortunately, the requirement for remote redirection surfaces quite often with protocols that aid nomadic computer users. Once the bond of static attachment has been severed, there is always the question whether control messages relevant to the mobile computer are authentic. For every network service that may be employed by nomadic users (for example, Mobile IP, DNS, or DHCP), authorization depends on some method (usually cryptographic) for verifying the identity of the requester. Location maintenance ranks high on the list of services requiring assurance of authorization, and the mobile node itself is usually considered (by mobile networking protocols) the highest authority on its location.

1.10 Example Architectures

Two examples of the foregoing abstract model are briefly described in this section. The first is Mobile IP (Perkins 1996b), as defined by the Internet Engineering Task Force (*IETF*). The second is a previous version of Mobile IP experimentally defined by researchers at Columbia University (Ioannidis and Maguire 1993), which has had substantial effects on the evolution of the IETF protocol.

1.10.1 Architectural Model of the IETF Protocol

In the IETF Mobile IP protocol, the LD is present at the same node on the home network that implements the readdressing function. The readdressing node on the home network is called the *home agent*. Correspondingly, a *foreign agent* fulfills the inverse readdressing function when the datagram is delivered to the care-of address. The intention is that the care-of address is owned by the foreign agent, and after the inverse readdressing function is performed, the foreign agent delivers the resulting datagram to the mobile node.

However, for sufficiently capable mobile nodes, it is quite reasonable to dispense with the foreign agents and allow the mobile nodes to perform the inverse readdressing function themselves. Notice that this also requires the mobile nodes to be able to acquire a suitable care-of address by some means. The only real constraint is that the care-of address be appropriate to the network to which the mobile node is currently attached, because the datagram cannot be delivered to the current location of the mobile node unless the care-of address is appropriate for that current location. One suitable mechanism (DHCP) by which a mobile node can acquire such a care-of address is detailed in Chapter 9.

Foreign agents can use a single care-of address to serve a number of mobile nodes. In contrast, mobile nodes that acquire their own care-of addresses will each require distinct addresses. This introduces the new requirement for further managing the multiplicity of care-of addresses at each network to which a mobile node might wish to attach.

Figure 1.6 shows an overview of the IETF protocol, with functions labeled according to the abstract model.

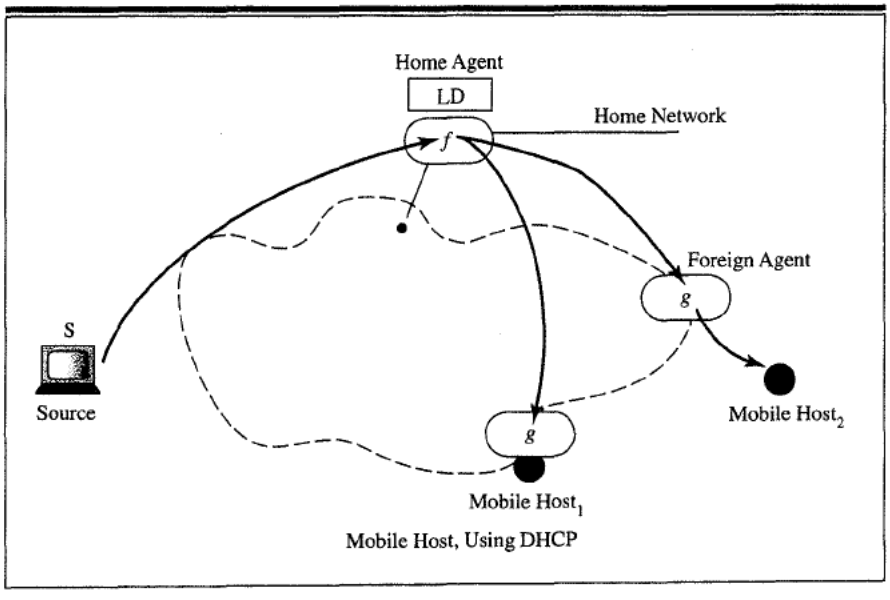

Figure 1.6 IETF Mobile IP proposal.

1.10.2 Columbia Mobile IP

Researchers at Columbia University were among the first to begin experiments in mobile networking. They aimed to provide campus mobility for mobile nodes, partially as an outgrowth of the Student Electronic Notebook (*SEN*) project. Columbia's

Mobile IP (Ioannidis, Duchamp, and Maguire 1991, Ioannidis and Maguire 1993) relied on the configuration of a collection of *mobile support routers* (*MSRs*) that conspired to create a *mobile subnet* (comparable to a home network, but having no physical instantiation) of IP addresses administered for use by the mobile nodes. As mobile nodes moved, they detected *beacons* emitted by the MSRs according to the Mobile Internet Control Protocol (*MICP*), comparable to ICMP, which was the protocol by which the beacons were delivered.

As mobile nodes moved from place to place, they informed their current MSR about their needs and requested that the current MSR inform their previous MSR of their movement. In this way all MSRs could remain up to date regarding the movement of the mobile node. The MSRs communicated by way of a new multicast address, which they had to join. See Figure 1.7 for an illustration of the Columbia protocol.

In terms of the abstract model, several functions in the Columbia protocol were distributed to all the cooperating MSRs supporting the mobile subnet. In particular, each MSR performed the following functions:

- Location directory
- Forward address redirection
- Inverse address translation

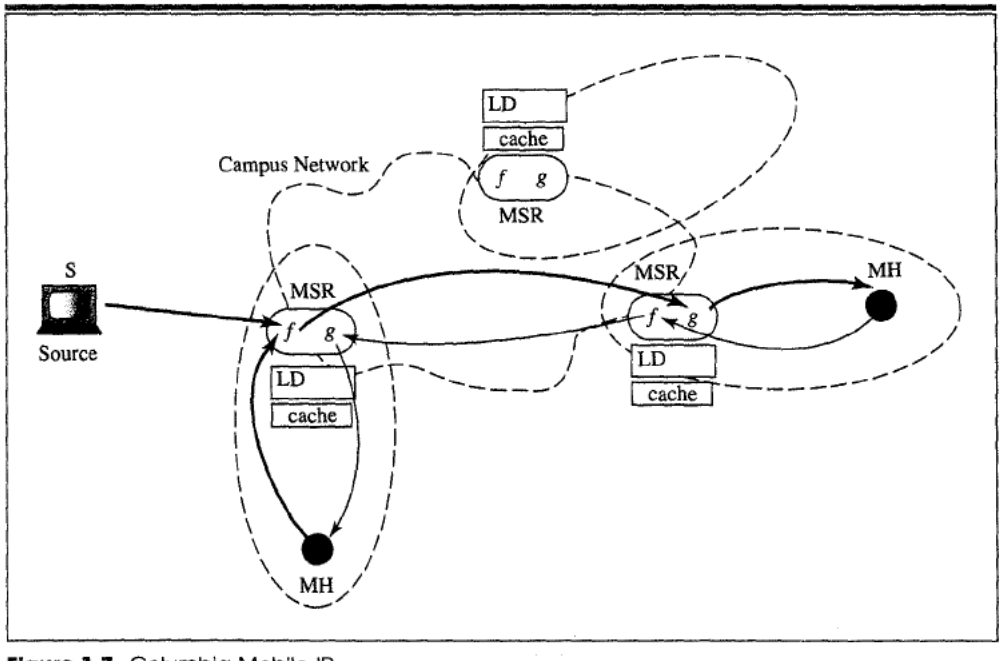

Figure 1.7 Columbia Mobile IP.

The Columbia protocol was very successful for many reasons, including its

- Elegant and simple design
- Symmetric architecture
- Low system impact at small-scale deployment
- Efficient implementation
- Historical precedence compared with other freely available implementation
- WaveLAN wireless interface
- Mach operating system platform

Since the mobile subnet was effectively a virtual subnet, no support for existing hosts on the mobile subnet was necessary. The Columbia protocol also served as the basis for numerous other research efforts into mobile networking at Rutgers and Brown universities, as well as further efforts at Columbia during the early 1990s. Especially important were their first attempts to provide network-layer security (Ioannidis and Blaze 1993). Concerns about scalability, however, drove attempts to avoid distributing the functions to a variable population of symmetric agents maintaining location information for the mobile nodes. With centralized location information, the use of multicast was no longer warranted, gaining possibly further improvements in scalability. This was especially true given the insufficient multicast protocol availability and deployment of the times. See Miles and Skellern (1993) for additional points of comparison between the IETF protocol and the Columbia protocol.

1.11 Where Mobile Networking Fits

This book explores ways to provide for node mobility by making the appropriate modifications to the networking protocol (IP) layer. One might ask why the networking layer should be the layer chosen to implement support for mobility when in fact it is possible to implement mobility at other layers. Socket support can be written so that applications using the augmented sockets work well (Bhagwat and Maltz 1997). TCP can be changed so that the TCP connection is broken into two parts (Indirect TCP), and the connection state passed off from one support station to another as the mobile unit moves. Other strategies are also possible. But providing mobility support at the IP layer fits most naturally, because as previously mentioned the mobility problem can be transformed into a routing problem in a natural way. This naturalness is evident in the simplicity of the protocol and the relatively small amount of code needed to implement the necessary changes to the route table handling at the home agent and foreign agent. Even so, mobility has other effects on protocols at every level of the network protocol stack, as described in the following sections.

1.11.1 Physical- and Link-Layer Protocols

The physical- and link-layer protocols have received the most attention (IEEE 802.11 Committee 1997, Rappaport 1996) for the purposes of mobile networking. This has been driven largely by the needs of mobile voice communications and military applications. Mobile networking needs that have been solved (with various degrees of success) below the network layer include:

- Adaptive error correction
- Low detectability
- Data compression
- Data encryption
- Power minimization
- Ad hoc networking
- Isochronous communications

See the standard by the IEEE 802.11 committee (1997) for a detailed consideration of such techniques. Managing ad hoc networks below the network layer seems to be a mistake, since routing protocols are inevitably involved in the maintenance of any dynamic topology of interconnections between the nodes populating the ad hoc network. Moreover, operations such as compression and encryption must be coordinated with higher level control software usually operating at the transport or application level. Lastly, protocols supporting *isochronous communications* (communications for which tight delay bounds and reliable bandwidth are needed) are not fully developed at the time of the writing of this book. It seems very likely that there will have to be close coordination between link management and the needs of the application programs (or transport protocol mechanisms) to have a sensible approach for providing isochronous communications capabilities. Thus, it seems premature to build such mechanisms into the link-layer protocols before the higher level needs are better understood.

The mobile node may use link-layer mechanisms to decide that its point of attachment has changed. These mechanisms are specific to the particular link-layer technology. Typically when the mobile node detects a change in its point of attachment by such means it is not necessary for the mobile node also to determine the subnet prefix for the new point of attachment. This is good, because the broadcast range of wireless subnets may not be very well defined. For cases when subnets are relevant when determining the point of attachment, extreme care has to be exercised in the way that subnet numbers are associated with the various wireless cells. For instance, if two wireless cells overlap, they may be unable to share the

same subnet prefix. A mobile node moving from one cell into another partially overlapping cell would almost always consider them to be two separate subnets as far as Mobile IP is concerned, unless link-layer mechanisms are in use such as those defined for the Institute of Electrical and Electronic Engineers (*IEEE*) 802.11 (1994) or cellular digital packet data (*CDPD*) (CDPD Consortium 1993).

When making comparisons to identify the nature of its current point of attachment, the mobile node should first attempt to specify its home address so that if the mobile node is attaching to its home network, the unrouted link will function correctly. If a transient IP address is dynamically assigned to the mobile node, and the mobile node is capable of supporting a colocated care-of address, the mobile node may register and use the transient address as a colocated care-of address.

1.11.2 TCP Considerations

TCP Timers

Most hosts and routers that implement TCP/IP do not permit easy configuration of the TCP timer values. When high-delay (for example, SATCOM) or low-bandwidth (for example, high-frequency radio) links are in use, the default TCP timer values in many systems may cause retransmissions or timeouts, even when the link and network are actually operating properly with greater than usual delays because of the medium in use. This can cause an inability to create or maintain TCP connections over such links, and can also cause unneeded retransmissions that consume already scarce bandwidth. Hopefully, mobility-aware vendors will begin to make TCP timers more configurable. Vendors of systems designed for the mobile computing market may have to pick default timer values more suited to low-bandwidth, high-delay links; otherwise, users of mobile nodes may have to be sensitive to the possibility of timer-related difficulties.

TCP Congestion Management

Mobile nodes often use wireless media, which are more likely to introduce errors, effectively causing more packets to be dropped. This introduces a conflict with the mechanisms for congestion management found in modern versions of TCP (Comer 1991, Stevens 1997). Currently, when a packet is dropped, the TCP implementation at the *correspondent node* (that is, the node passing information back and forth to the mobile node) is likely to react as if there was network congestion. This initiates the slow-start (Stevens 1997) mechanisms designed for controlling that problem. However, these mechanisms are inappropriate for overcoming errors introduced by the links themselves and they have the effect of magnifying the discontinuity introduced by the dropped packet. This problem has been analyzed by Caceres and Iftode (1995). There is no easy solution available, and certainly no solution

is likely to be installed soon on a majority of IP nodes. This problem illustrates that providing performance transparency to mobile nodes involves understanding mechanisms outside the network layer (Kleinrock 1995). It also emphasizes the need to avoid designs that systematically drop packets. Such designs might otherwise be considered favorably when making engineering trade-offs.

1.12 Middleware Components

Once sufficient capabilities are available in the network protocol stacks to support mobile networking, the need for other capabilities will become more evident. There are a great many services needed by nomadic computer users that have not received much attention yet because the lack of basic protocol support has made the further requirements somewhat of a moot point. These *nomadic services* are likely to be arranged as a set of *middleware* components; that is, service modules that run external to the basic network protocols, but which are viewed as system services by the applications that need to invoke the services to support nomadic users.

Several types of middleware components have been identified as part of a *nomadic architecture* being developed by the Cross Industry Working Team (*XIWT*) group effort for nomadic computing (Corporation for National Research Initiatives 1994). The following list is far from complete, but at least seems likely to be part of any future list. Moreover, further consideration of how the components interact is likely to produce additional needs and the identification of specific new middleware components. Identified middleware components include those that enable the

- Ability to locate network resources
- Ability to adapt to changing link conditions
- Management of profile options and context awareness
- Configuration of local environmental agents

In this section some of the characteristics and uses of these middleware components are briefly considered. Along a somewhat different track, one can consider proxy services and intelligent agents, which can also provide nomadic services, but which are usually too self-contained to be considered middleware. Middleware services are likely to be viewed as part of the operating system by nomadic applications, whereas intelligent agents would be considered entirely separate entities.

1.12.1 Service Location Protocol

The Service Location Protocol (*SLP*) (Veizades et al. 1997) is a new Proposed Standard protocol for discovering and contacting network services, thereby avoiding the continued need for static preconfiguration of all such services that currently

plagues mobile computer users. SLP defines the protocol actions required for *user agents* (UAs) to determine the network access points dynamically for network services, which themselves are known by their corresponding *service agents* (SAs). In the most general setting, SAs advertise their services and presence by establishing database entries with a suitable (usually local) *directory agent* (DA). SLP defines the messages exchanged by UAs, SAs, and DAs. In addition, each specific service will define the access mechanisms and descriptions (*attributes* and *keywords*) by which it is made known to UAs. SLP defines the ways that the agents communicate their needs and offerings, not the service-specific descriptions.

SLP may be generally described as specifying the following kinds of operations:

- Service request and reply
- Service registration
- DA discovery
- Attribute enumeration

Each UA uses a *service request* to obtain a *service reply*, which indicates the network address of the appropriate SA. Such service request messages ask for a generically named service, and a specific instance of that service is expected in return. In addition to requesting the generic service, a user will typically need to specify certain desired characteristics that must be available from the SA. For instance, a UA requesting a printer service might need to be sure that the printer can handle PostScript files.

To simplify administration and enhance the scalability of the protocols, a DA should be made available to user clients, because it manages the database of known services for all users. The DA accepts *service advertisements* from service agents; the advertisement messages are expected to contain enough descriptive data (attributes and keywords) to enable the DA to determine when the advertised service will meet the needs specified in the service requests.

One of the main contributions of SLP is to define a general set of mechanisms for discovering DAs. Both UAs and SAs need to identify any DAs that might be available to serve them. Managing the DAs will likely be the main administrative burden imposed by SLP. UAs and SAs will naturally evolve to use DAs as they become available, so that the UAs and SAs can be deployed without the need for any preconfiguration. For instance, DAs can be discovered by use of DHCP, so that the entire administrative load of DA discovery then centers on the ways to classify and organize them in the DHCP server database.

Lastly, SLP defines mechanisms by which UAs can collect sets of available services and service attributes that are available locally. This presumably would be under user control, and the user would be able to satisfy application requirements

interactively according to the set of available services that is discovered. In general, SLP has been defined so that it is equally well suited for interactive or noninteractive use.

The benefits provided by SLP for nomadic users should be clear. Nomads are quite likely to be inconvenienced by any need to perform lengthy configuration sequences manually. Both the information-gathering and administrative aspects of the configuration procedures can be arbitrarily difficult and error prone. Any doubt on this point can be banished by a moment's reflection on current means of establishing printer service or groupware connections, finding mountable file systems, or operating dial-up services. All of these services can be simplified by the use of SLP, and all are of interest to nomadic users. As the future unfolds, and the network truly does replace the computer as the center of computing interest, network services are likely to proliferate wildly, as the Internet already has.

1.12.2 Link Adaptivity

Recent investigation shows that the optimal network response to user applications depends heavily on the conditions affecting the link between the computer and the network to which it is attached. For instance, a Web user would like to have new pages presented in a timely fashion no matter what the current link conditions. If speed of presentation requires the replacement of high-resolution graphics with less fascinating markers, then that trade-off is almost always preferred by users experiencing network congestion. Moreover, since protocol implementations can have an internal state indicating congested situations, users can rightfully expect that the Web applications should be able to adjust to link conditions dynamically and make the appropriate trade-offs automatically.

Unfortunately, current network protocol support does not usually offer this capability. Indeed, the problem is multifaceted and will require progress along several fronts:

- Link information will have to be classified so that it can be presented according to some standardized format.
- A standardized application programming interface (*API*) is needed, analogous to the near-ubiquitous and famously useful *sockets API* originated at the University of California at Berkeley.
- Link information at all network protocol levels (physical interface, link-layer, IP, and transport protocols) will have to be made available to the API routines, again indicating the need for a standardized programming interface at the systems programming level.

Some link characteristics of interest to nomadic-aware applications include

- Available bandwidth
- Latency
- Cost (and cost structure)
- Availability of security and privacy
- Error rate
- Quality of service promises

The emergence of wireless interfaces has had the effect of emphasizing the need for this information, since it is particularly wireless communications that make it more difficult to obtain any sort of static characterization of the relevant parameters. Recent work in defining extensions to the Network Device Interface Standard (*NDIS*) device interface (*NetDev* (Stardust Technologies 1996)) represents first efforts to provide link adaptivity information. Designers are also referred to work done at Carnegie Mellon University to classify the kinds of link information needed according to various dimensions of *fidelity* (Satyanarayanan 1996).

1.12.3 Profile Management

As the population of nomadic users grows, mobile computers will be seen in an ever-increasing variety of social situations. It will become more and more important to exercise control over the way that applications respond, depending on the social situation surrounding the nomadic use of the application. This differs from link adaptivity in two important respects:

1. The required information is not easily available by inspecting the internal state of the network protocol engines.
2. The same physical location can correspond to a number of different social situations, depending on the time of day, time of year, identity of the nomadic user, and indeed an indeterminable number of other factors as diverse as society itself.

The middleware component presumed to provide assistance in managing application response in accordance to the context in which the application is used is called the *profile manager*. The profile manager is expected to maintain a set of application-specific profiles and to retrieve stanzas of the profile in accordance with user preferences. The user preferences could be established interactively or by reference to various *environmental variables*. Today, environmental variables are

commonly set whenever the computer is initialized or rebooted. But to fulfill the needs of nomadic users it is anticipated that such environmental information will also become much more dynamic.

1.12.4 Environment Manager

One can visualize the need for yet another middleware component—an *environment manager*—that would manage a set of dynamic environmental variables in response to signals detected from active agents within the effective range in the nomad's local environment. Work with Active Badges (Want et al. 1992) already indicates a strong need for adapting the behavior of various applications in response to the presence (or absence) of active environmental agents, and in accordance with the state of those agents. For instance, in a conference room a nomadic user would most likely wish for paging applications to operate differently than if the user were alone. This adaptation on the basis of external, non-link-related environmental factors represents a new challenge for nomadic-aware applications that has only begun to be addressed.

1.13 Proxies versus Mobile-aware Applications

Nomadic users will create a new need for proxy services that can help perform appropriate translations or other support services for mobile computers (Samara et al. 1997). For instance, a mobile computer that most easily understands a particular kind of video encoding may still wish to receive multicasts from sources of video data that use different encodings. In particular, certain hardware platforms accelerate the display of video data encoded in particular formats (Amir, McCanne, and Zhang 1995), and some compression algorithms offer much higher and effective bandwidths if utilized correctly in the hardware. Computers that do not employ compatible hardware may well wish to avoid the transmission of video data in the wrong compression format or encoding.

Alternatively, proxies can be useful to perform authentication for mobile computers, which are sometimes performance limited. Today's software encryption techniques, to achieve the cryptographic strength needed to protect digital commercial transactions, may require a computational speed unavailable in the low-power devices characterizing *personal digital assistants* (PDAs), which are designed for simple appointment management, expense accounting, and other pocket calendar tasks for which efficient battery power management is essential.

Another example of the power of proxy agents may be seen with the proposed *intelligent agents* (Kraster 1995) that may roam the World Wide Web according to a schedule mostly independent of the actions of the nomadic user. When the nomadic user chooses to contact the intelligent agent, or when the intelligent agent can establish connectivity to indicate the need for a transaction, the nomad will then

acquire and process any buffered (stored) data that may have become available. Proxy agents may also be able to store buffered data for mobile users in case data is lost during transmission over a wireless medium.

For all of these reasons—special hardware capabilities, additional processing power, availability of proprietary software, overcoming connectivity problems, and buffering data to correct errors—proxy services and intelligent agents are likely to be designed and made available for nomadic users of mobile computers.

It should be noted that the widespread use of proxies to support nomadic computer users may introduce a new kind of rigidity in the overall network architecture. With proxies, a mobile node always has to remain compatible with the software running on the proxy service. Since a proxy server is likely to be assisting a large and diverse population of mobile clients, it seems unlikely that the server and the clients could all be updated together. Thus, the presence of proxies adds to the number of software components that need upgrades, and could make it more awkward to update software on the mobile computers themselves. Perhaps the proxy server will be maintained efficiently enough so that many different software versions of the same service can be invoked depending on the needs of particular clients. This effect of proxy service should be watched carefully and compared against the possible benefits of allowing the mobile node to operate independently of any proxy. Use of SLP (Section 1.12.1) to locate proxies with special features may improve the chances for their widespread development.

Lastly, note that the forwarding of datagrams through a proxy server before they are delivered to a mobile node represents another routing irregularity, and another possible point of failure in a data transaction. These disadvantages will, hopefully, be more than compensated for by the increased computing power of the proxy service.

1.14 Summary

This chapter explained the need for mobile networking to support the requirements of today's new class of Internet users as they roam about with sophisticated laptop computers and digital wireless data communication devices. The exponential growth of the Internet and the inexorable increase in native computing power of laptop computers have brought the need for mobile networking into sharp focus. As network services proliferate and become available ubiquitously, every network device will take advantage of mobile networking technology to offer maximum flexibility to the customers needing those devices.

The problem solved by Mobile IP is the use of IP addresses for both identifying an IP node and selecting the route to that node. An abstract model was shown that clarifies the need for manipulating the care-of address, which provides the level of

addressing indirection needed for routing to the mobile node, while still allowing the mobile node to use its home address, by which it is identified.

Mobile IP, while useful as a general technique to solve a number of problems caused by mobility, nevertheless does not solve all the problems. In fact, some problems that are obvious while roaming with a mobile computer were previously unrecognized, especially those having to do with reconfiguring network access to resources and dealing with variable network connections. New features will be demanded for the further simplification of the nomadic computer user's location-independent interaction with the Internet and its information space.

CHAPTER TWO

Mobile IP Overview

This chapter discusses the main concepts and operations of the IETF Mobile IP protocol. The basic protocol procedures fall into the following areas:

- Advertisement
- Registration
- Tunneling

The functional entities performing these procedures are illustrated, along with the typical interactions among them. Two different ways to acquire care-of addresses are described. Lastly, a brief introduction to the IETF, which hosts the Mobile IP Working Group, is presented to establish an important context that clarifies the procedure by which Mobile IP has ultimately been promoted as a Proposed Standard protocol for the Internet.

2.1 What Is Mobile IP?

Mobile IP is a modification to IP that allows nodes to continue to receive datagrams no matter where they happen to be attached to the Internet. It involves some additional control messages that allow the IP nodes involved to manage their IP routing tables reliably. Scalability has been a dominant design factor during the development of Mobile IP, because in the future a high percentage of the nodes attached to the Internet will be capable of mobility.

As explained in the last chapter, IP assumes that a node's network address uniquely identifies the node's point of attachment to the Internet. Therefore, a node must be located on the network indicated by its IP address to receive datagrams destined to it; otherwise, datagrams destined to the node would be undeliverable. Without Mobile IP, one of the two following mechanisms typically must be employed for a node to change its point of attachment without losing its ability to communicate:

1. The node must change its IP address whenever it changes its point of attachment.

2. Host-specific routes must be propagated throughout the relevant portion of the Internet routing infrastructure.

Both of these alternatives are plainly unacceptable in the general case. The first makes it impossible for a node to maintain transport and higher layer connections when the node changes location. The second has obvious and severe scaling problems that are especially relevant considering the explosive growth in sales of notebook (mobile) computers.

Mobile IP was devised to meet the following goals for mobile nodes that do not *move* (that is, change their point of attachment to the Internet) more frequently than once per second. Even so, the protocol is likely to work well until the frequency of movement of the mobile node begins to approach the round-trip time for Mobile IP protocol control messages. The following five characteristics should be considered baseline requirements to be satisfied by any candidate for a Mobile IP protocol:

1. A mobile node must be able to communicate with other nodes after changing its link-layer point of attachment to the Internet, yet without changing its IP address.
2. A mobile node must be able to communicate with other nodes that do not implement Mobile IP. No protocol enhancements are required in hosts or routers unless they are performing the functions of one or more of the new architectural entities introduced in Section 2.2.
3. All messages used to transmit information to another node about the location of a mobile node must be *authenticated* to protect against remote redirection attacks.
4. The link by which a mobile node is directly attached to the Internet may often be a wireless link. This link may thus have a substantially lower bandwidth and higher error rate than traditional wired networks. Moreover, mobile nodes are likely to be battery powered, and minimizing power consumption is important. Therefore, the number of administrative messages sent over the link by which a mobile node is directly attached to the Internet should be minimized, and the size of these messages should be kept as small as possible.
5. Mobile IP must place no additional constraints on the assignment of IP addresses. That is, a mobile node can be assigned an IP address by the organization that owns the machine, as is done with any other protocol engine administered by that organization. In particular, the address does not have to belong to any globally constrained range of addresses.

Mobile IP is intended to enable nodes to move from one IP subnet to another. It is just as suitable for mobility across heterogeneous media as it is for mobility across homogeneous media. That is, Mobile IP facilitates node movement from

one Ethernet segment to another as well as accommodates node movement from an Ethernet segment to a wireless LAN, as long as the mobile node's IP address remains the same after such a movement.

One can think of Mobile IP as solving the *macro* mobility management problem. As long as node movement does not occur between points of attachment on different IP subnets, link-layer mechanisms for mobility (that is, link-layer handoff) may offer alternative solutions with different engineering trade-offs compared with Mobile IP. For instance, the IEEE has recently standardized such an alternative solution for wireless mobility in their IEEE 802.11 committee (1997).

Note that Mobile IP does not place any requirement on the layer-2 (link-layer) operation of a mobile node. This means that it is equally suitable to manage the mobility of a node no matter what the physical nature of the node's link to the Internet. Mobile IP works as well for nodes moving from one Ethernet to another as it does for nodes moving from one base station to another with a radio connection, as long as the link itself is established equally well. Some layer-2 protocols handle node mobility in restricted ways. Mobile IP can still work with those layer-2 protocols to provide wider area mobility, since it is difficult for layer-2 protocols to provide mobility across IP subnets.

2.2 Terminology

Mobile IP introduces the following new functional entities:

Mobile node—A mobile node is a host or router that changes its point of attachment from one network or subnetwork to another. A mobile node may change its location without changing its IP address. It may continue to communicate with other Internet nodes at any location using its (constant) IP address, assuming link-layer connectivity to a point of attachment is available.

Home agent—A home agent is a router on a mobile node's home network that *tunnels* datagrams for delivery to the mobile node when it is away from home and maintains current location information for the mobile node.

Foreign agent—A foreign agent is a router on a mobile node's *visited network* that provides routing services to the mobile node while registered. The foreign agent detunnels and delivers datagrams to the mobile node that were tunneled by the mobile node's home agent. The foreign agent may always be selected as a default router by registered mobile nodes.

A mobile node is given a long-term IP address on a home network. This home address is treated administratively just like a permanent IP address provided to a stationary host. When away from its home network, a care-of address is associated with the mobile node and reflects the mobile node's current point of attachment.

The mobile node uses its home address as the source address of all IP datagrams that it sends, except during registration if it happens to acquire another IP address (as described in Section 4.6.1).

Looking back at Figure 1.6, recall that the home agent should be thought of as the combination of an LD for the mobile node's care-of address and a readdressing/redirecting function f for packets that arrive at the home network. Conversely, the foreign agent should be considered to be the inverse readdressing function g to restore the datagram to its original form after manipulation by the home agent, followed by delivery to the mobile node.

As a matter of adherence to the protocol specification, in this book the phrase *is required to* means that the implementor is required to abide by the stated condition to be considered in compliance. The word *should*, on the other hand, indicates that some implementations may omit the suggested operation or condition; however, this should only be done in particular cases when the designer of the implementation is well aware of all the implications of the omission. The Mobile IP protocol was specified with the expectation that implementations would in fact support all the features indicated by *should*.

On the negative side, the phrase *not allowed* means that an implementation that permits the indicated condition or operation does *not* comply with the protocol specification. The phrase *should not* means that any designer should be very wary of allowing the operation or condition under consideration. Lastly, the word *may* usually means that the specified operation or condition is purely optional and is allowed but not required. All implementation entities are required to handle messages requesting optional features without malfunctioning, even if the entities cannot support those features.

2.3 Protocol Overview

Mobile IP is, in essence, a way of doing three relatively separate functions:

1. *Agent discovery*—Home agents and foreign agents may advertise their availability on each link for which they provide service. A newly arrived mobile node can send a solicitation on the link to learn if any prospective agents are present.
2. *Registration*—When the mobile node is away from home, it registers its care-of address with its home agent. Depending on its method of attachment, the mobile node will register either directly with its home agent or through a foreign agent, which forwards the registration to the home agent.
3. *Tunneling*—In order for datagrams to be delivered to the mobile node when it is away from home, the home agent has to *tunnel* the datagrams to the care-of address.

When away from home, Mobile IP uses protocol tunneling to hide a mobile node's home address from intervening routers between its home network and its current location. The tunnel terminates at the mobile node's care-of address. The care-of address must be an address to which datagrams can be delivered via conventional IP routing. At the care-of address, the original datagram is removed from the tunnel and delivered to the mobile node.

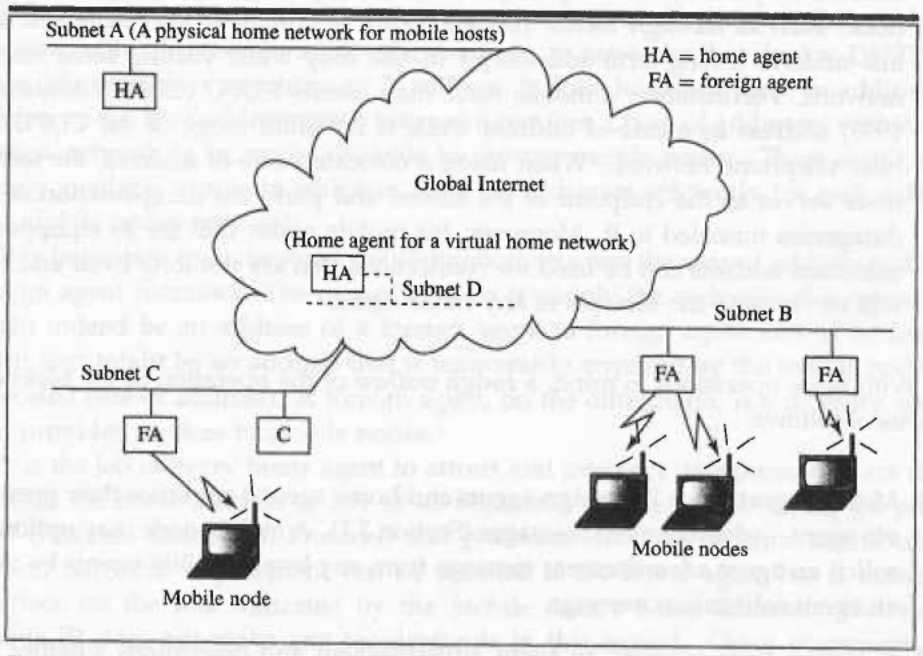

Figure 2.1 Mobile IP.

An overall illustration of the entities of Mobile IP and several home and *foreign networks* is shown in Figure 2.1. In the diagram there are two foreign networks, B and C, with foreign agents; two home networks, A and D, with home agents; and mobile nodes that are attached to the various foreign networks by way of radio and infrared attachments. The tunnels go from the home agents, across the global Internet, and finally arrive at the foreign agents for final delivery.

Mobile IP provides two ways to acquire a care-of address:

1. A foreign agent care-of address is a care-of address provided by a foreign agent through its *agent advertisement* messages. In this case the care-of address is an IP address of the foreign agent. In this mode, the foreign agent is the endpoint of the tunnel and, on receiving tunneled datagrams, decapsulates them and delivers the inner datagram to the mobile node. This mode of acquisition is

advantageous because it allows many mobile nodes to share the same care-of address and therefore does not place unnecessary demands on the already limited Internet Protocol version 4 (*IPv4*) address space.

2. A colocated care-of address is a care-of address acquired by the mobile node as a local IP address through some external means, which the mobile node then associates with one of its own network interfaces. The address may be dynamically acquired as a temporary address by the mobile node, such as through DHCP (Droms 1993), or it may be owned by the mobile node as a long-term address for its use only while visiting some foreign network. For instance, a mobile node may use its CDPD (CDPD Consortium 1993) address as a care-of address while it is within range of the CDPD cellular telephone network. When using a colocated care-of address, the mobile node serves as the endpoint of the tunnel and performs decapsulation of the datagrams tunneled to it. Moreover, for mobile nodes that are so equipped, a colocated address can be used for connections that are not long lived and thus will never need the services of any home agent.

With these operations in mind, a rough outline of the operation of the Mobile IP protocol follows:

1. *Mobility agents* (that is, foreign agents and home agents) advertise their presence via agent—advertisement messages (Section 3.1). A mobile node may optionally solicit an agent advertisement message from any local mobility agents by using an agent solicitation message.
2. A mobile node receives an agent advertisement and determines whether it is on its home network or a foreign network.
3. When the mobile node detects that it is located on its home network, it operates without mobility services. If returning to its home network from being registered elsewhere, the mobile node deregisters with its home agent through a variation of the normal registration process.
4. When a mobile node detects that it has moved to a foreign network, it obtains a care-of address on the foreign network. The care-of address can either be a foreign agent care-of address or a colocated care-of address.
5. The mobile node, operating away from home, then registers its new care-of address with its home agent through the exchange of a registration request and registration reply message, possibly by way of a foreign agent (Sections 4.3, 4.4).
6. Datagrams sent to the mobile node's home address are intercepted by its home agent, tunneled by the home agent to the mobile node's care-of address, received

at the tunnel endpoint (either at a foreign agent or at the mobile node itself), and finally delivered to the mobile node (Chapter 5, Section 5.9.3).

7. In the reverse direction, datagrams sent by the mobile node may be delivered to their destination using standard IP routing mechanisms, without necessarily passing through the home agent.

Using a colocated care-of address has the advantage of allowing a mobile node to function without a foreign agent—for example, in networks that deploy DHCP or some other means of acquiring an IP address. It does, however, place an additional burden on the IPv4 address space because it requires a pool of addresses within the foreign network to be made available to visiting mobile nodes. There aren't any widely available means to maintain pools of addresses efficiently for each subnet that mobile nodes may visit.

It is important to understand the distinction between the care-of address and the foreign agent functions. The care-of address is simply the endpoint of a tunnel. It might indeed be an address of a foreign agent (a foreign agent care-of address), but it also might be an address that is temporarily acquired by the mobile node (a colocated care-of address). A foreign agent, on the other hand, is a mobility agent that provides services to mobile nodes.

It is the job of every home agent to attract and intercept datagrams that are destined to the home address of any of its registered mobile nodes. Using the *proxy ARP* (Address Resolution Protocol) and *gratuitous ARP* mechanisms described in Section 5.13, this requirement can be satisfied if the home agent has a network interface on the link indicated by the mobile node's home address. However, Mobile IP does not make any requirements in this regard. Other placements of the home agent relative to the mobile node's home location are possible, using other mechanisms for intercepting datagrams destined to the mobile node's home address. Three obvious candidates for placement of the home agent on the home network are illustrated in Figure 2.2, as follows:

A = a home agent as a separate system on the home network,

B = a home agent included with a router to the home network, and

C = a virtual home network.

Similarly, a mobile node and a prospective or current foreign agent must be able to exchange datagrams without relying on standard IP routing mechanisms, which make forwarding decisions based on the network prefix of the destination address in the IP header. This requirement can be satisfied if the foreign agent and the

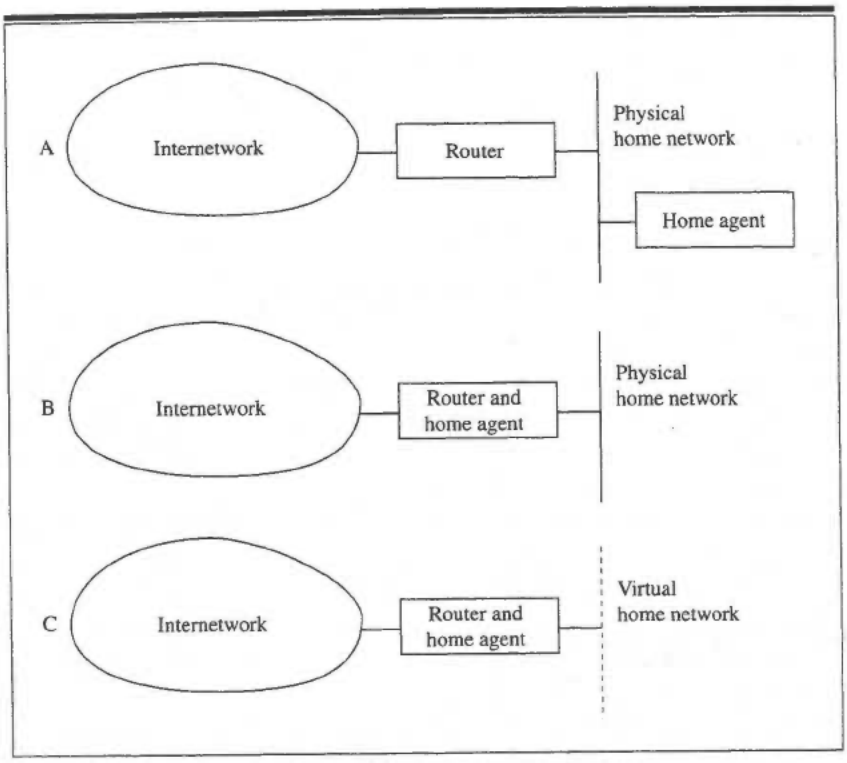

Figure 2.2 Ways to put a home agent on a home network.

visiting mobile node have an interface on the same link. In this case, the mobile node and the foreign agent simply bypass their normal IP routing mechanism when sending datagrams to each other, addressing the underlying link-layer packets to their respective *link-layer addresses*. Mobile IP allows any placement of the foreign agent relative to the mobile node, using other mechanisms to exchange datagrams between these nodes, as long as the basic protocol is followed.

Of course, if a mobile node is using a colocated care-of address (as described at the beginning of this section), the mobile node is required to be located on the link identified by the network prefix of this care-of address. Otherwise, datagrams destined to the care-of address would be undeliverable.

Figure 2.3 illustrates the routing of datagrams to and from a mobile node away from home, once the mobile node has registered with its home agent. In this figure the mobile node is using a foreign agent care-of address as follows:

1. A datagram to the mobile node arrives on the home network via standard IP routing.
2. The datagram is intercepted by the home agent and is tunneled to the care-of address.

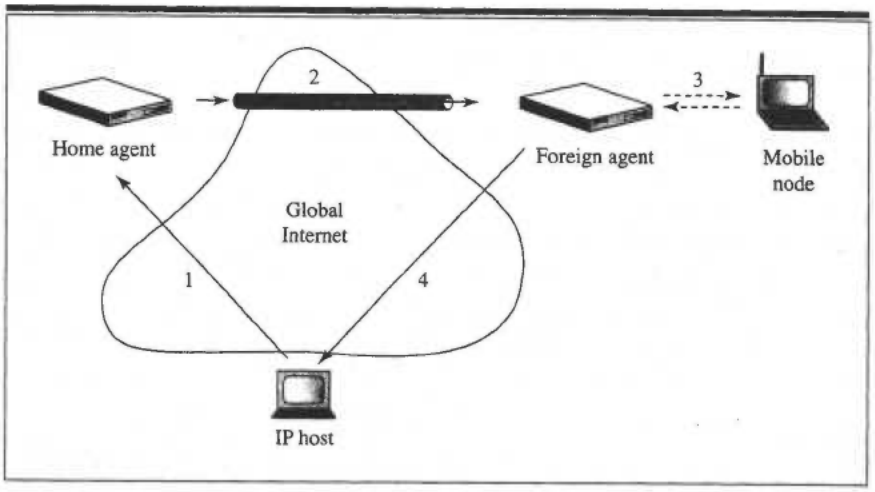

Figure 2.3 Mobile IP datagram flow.

3. The datagram is detunneled and delivered to the mobile node.
4. For datagrams sent by the mobile node, standard IP routing delivers each datagram to its destination. In Figure 2.3, the foreign agent is the mobile node's default router.

2.4 Message Format and Protocol Extensibility

To handle registration, Mobile IP defines a set of new control messages sent with UDP (Postel 1980) using well-known port number 434. Currently, the following two message types are defined:

1 Registration request

3 Registration reply

Up-to-date values for the message types for Mobile IP control messages are specified in the most recent *Assigned Numbers* (Reynolds and Postel 1994).

For agent discovery, Mobile IP modifies the existing router advertisement and router solicitation messages defined for ICMP router discovery (Deering 1991), as described in Section 3.3.

Mobile IP defines a general extension mechanism to allow optional information to be carried by Mobile IP control messages or by ICMP router discovery messages. Each of these extensions (with one exception, the *pad extension*) is encoded in what is

conventionally called the *type-length-value* (*TLV*) format shown in Figure 2.4, where the *value* is the data following the *length*.

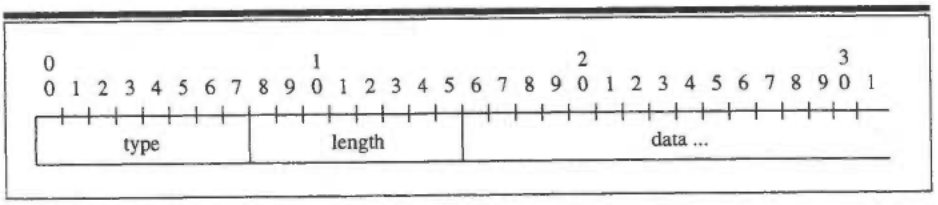

Figure 2.4 The TLV extension format.

The *type* indicates the particular type of extension. The *length* of the extension, counted in bytes—or, more technically in octets, which are groups of 8 bits—does not include the type and length bytes, and may be zero or greater. The format of the data field is determined by the type and length fields. Extensions allow variable amounts of information to be carried within each message. The end of the list of extensions is determined by the total length of the IP datagram.

Two separately maintained sets of numbering spaces, from which extension type values are allocated, are used in Mobile IP. The first set consists of those extensions that may appear in Mobile IP control messages (those sent to and from UDP port number 434). Currently, the following types are defined for extensions appearing in Mobile IP registration messages:

32 Mobile—home authentication

33 Mobile—foreign authentication

34 Foreign—home authentication

The second set consists of those extensions that may appear in ICMP router discovery messages. Currently, Mobile IP defines the following types for such extensions:

0 One-byte padding (encoded with no length or data field)

16 Mobility agent advertisement

19 Prefix length

Each individual extension is described in detail later in a separate section. Up-to-date values for these extension type numbers are specified in the most recent list of *Assigned Numbers* (Reynolds and Postel 1994) from the Internet Assigned Numbers Authority (*IANA*).

Since these sets of extensions are independent, it is conceivable that two unrelated extensions that are defined at a later date could have identical type values. One of the extensions could be used only in Mobile IP control messages and the other only in ICMP router discovery messages.

The value of the extension number is important when trying to determine the correct disposition of unrecognized extensions. When an extension numbered in either of these sets within the range 0 through 127 is encountered but not recognized, the message containing that extension is required to be silently discarded. When an extension numbered in the range 128 through 255 is encountered but unrecognized, that particular extension is ignored, but the rest of the extensions and message data are still required to be processed. The length field of the extension is used to skip the data field in searching for the next extension.

2.5 Role of the IETF

Mobile IP has been standardized through the efforts of the Mobile IP Working Group organized under the jurisdiction of the IETF. The IETF is a collection of approximately 70 working groups, according to the last count. The working groups themselves are organized into a number of areas, and each area is supervised by an area director. The groups are created by petition, presented to the area director, often after a birds of a feather (*BOF*) session that gauges community interest. Each working group is also supervised by one or more working group chairpersons, and often a document editor is selected by the working group chair to facilitate the production of any Internet Drafts that are needed. The draft documents are intended to be working documents, subject to change at any time, and subject to expiration after being made available for six months in a collection of repositories (called *shadow directories*) maintained around the world. The documents are available free of charge and can be obtained by using a Web browser pointed at http://www.ietf.org, or by *anonymous File Transfer Protocol* (FTP) from the repositories, conventionally in a file system subdirectory named *internet-drafts*. The currently available shadow directories include

- ftp.is.co.za (Africa)
- nic.nordu.net (North Europe)
- ftp.nis.garr.it (South Europe)
- munnari.oz.au (Pacific Rim)
- ftp.ietf.org (US East Coast)
- ftp.isi.edu (US West Coast)

The history of the IETF helps to explain the evolution and final status of Mobile IP. The original working group was created by researchers Steve Deering (then with Xerox Palo Alto Research Center), and Chip Maguire, John Ioannides, and Dan Duchamp (then of Columbia University). The Columbia protocol described in Chapter 1 was already in the process of being finalized when researchers from other institutions began to show a high degree of interest in the working group. In particular, I had been working on an alternative design that concentrated the LD in a single entity (which has since evolved into the home agent). Other efforts, notably the Mobile Host Routing Protocol (*MHRP*) (Johnson 1994) and the Virtual Internet Protocol (*VIP*) (Teraoka and Tokoro 1993) by Sony were introduced and debated intensely. At some point, suitable terminology had finally come into use, but seemingly endless discussions ensued with little clear indication of which operations might end up in a deployable protocol. After orders from the area director and working group cochairs, proponents of various approaches began to find consensus and the working group at large determined that the deployment of Mobile IP should probably proceed in at least two stages. The first deployment would be the *base protocol*, which allows for operation with no changes to existing Internet computers, but which suffers from problems with suboptimal routing (remedied in Chapter 6). Later deployment questions were at that time thought to center around finding the best ways to modify existing computers to find better routes for mobile nodes. Now, however, even though the base protocol is standardized, deployment questions are centering around *firewall issues* (Section 7.1) instead of route optimization.

2.6 Summary

Mobile IP uses a straightforward protocol to supply the needed routing information to a mobile node's home agent so that it can do the work needed to redirect traffic from the home network to the care-of address. The protocol relies on the foreign agent to advertise its presence, and to relay registration messages back and forth between the mobile node and the home agent. The advertisements fit within extensions to the ICMP Router Advertisement protocol, and the registration messages are carried in UDP packets with retransmission specifications, avoiding some of the complexity of TCP.

The IETF Mobile IP Working Group has shepherded the protocol through the IETF processes until it is now finally a Proposed Standard. Future and continued work on the Mobile IP standard will proceed within the working group, and can be expected to define and refine other extensions to the basic ICMP messages and UDP registration messages that make up the base protocol. Some of the proposed extensions that are not yet standard are described later in this book.

CHAPTER THREE

Advertisement

This chapter presents the detailed message formats used by mobile computers and mobility agents to discover each other's presence. These messages are communicated by way of ICMP, according to the original protocol specification for *router discovery*. The router discovery protocol is, therefore, described in some detail so that this book is relatively self-contained. The extra information required to initiate the Mobile IP registration procedures is contained in message extensions to the Router Advertisement message, which is then called an *agent advertisement* message.

Agent discovery is the method by which a mobile node (1) determines whether it is currently connected to its home network or to a foreign network and (2) detects when it has moved from one network to another. This chapter describes the message formats and procedures by which mobile nodes, foreign agents, and home agents cooperate to realize agent discovery. The methods specified in this chapter also allow the mobile node to determine the care-of addresses offered by each foreign agent on any network to which it might connect.

3.1 Agent Solicitation and Discovery Mechanisms

An agent advertisement is formed by including a mobility agent advertisement extension (Section 3.3) in an ICMP Router Advertisement message (Section 3.2). An *agent solicitation message* is very similar in format to an ICMP router solicitation.

Agent advertisement and solicitation may not be necessary for link layers that already provide this functionality. The method by which mobile nodes establish link-layer connections with prospective agents can be different for each kind of link-layer. For instance, with Ethernet, nodes obey the proper framing and timing specification and obtain the appropriate IEEE 802.2 address for the destination (which can be handled by higher layers) (Metcalfe and Boggs 1976). For the typical wireless link layer, however, there are other considerations. These may include getting an appropriate radio channel, avoiding the *hidden terminal problem* (Bantz and Bauchot 1994) by using a reservation-based media access control (*MAC*) layer protocol (for instance, MACAW (Bharghavan et al 1994)), or any of several other

protocol problems that are specific to particular media. The procedures described in this chapter assume that such link-layer connectivity has already been established. Refer to other publications for details on how to establish the link (IEEE 802.11 Committee 1997).

No authentication is required for agent advertisement and agent solicitation messages. As far as Mobile IP is concerned, any agent that advertises its service and performs the needed functions to carry out the service is a bona fide mobility agent. This essentially allows any imposter to pretend to be a foreign agent, and the Mobile IP protocol has been designed with this fact in mind. Impersonating a home agent is typically more difficult given that the home agent and mobile node share a private *mobility security association*.

3.2 Router Discovery Protocol

Mobile IP extends ICMP router discovery as its primary mechanism for agent discovery. Therefore, it is important to understand the relevant details of ICMP router discovery, and a short explanation is included here for that purpose. The following material is taken from RFC 1256 (Deering 1991).

Hosts on a link typically must use the services of a directly attached router to deliver their datagrams to hosts on any other link. In fact, it is quite often the case that hosts send all such datagram traffic through a single router—the *default router*.

Determining the IP addresses of the locally attached router or routers was historically a matter for manual configuration. Later, efficient administrators typically ran locally developed programs to set up the router addresses as part of the machine configuration when the operating system was first loaded. Both of these strategies (especially the former) are likely to offer little help for the problems caused by reconfiguration, when a computer is moved or (less often) when a router is no longer available at the expected address.

Router discovery provides the means by which IP hosts can determine automatically the local routers' IP addresses and can monitor their continued presence. This is done by using two simple ICMP messages—one transmitted by the routers and another that may be transmitted by the hosts themselves. Since the router discovery protocol included enough features to allow Mobile IP hosts to discover foreign agents, it was used as the basis for the mobility agent discovery mechanisms described in this chapter. At the time, it seemed reasonable to avoid the creation of new protocols when previously existing protocols might serve just as well. If this decision were revisited today it would almost certainly be reversed, and a new protocol would be created just for the purposes of Mobile IP (see Section 3.7).

More recently it has become feasible to configure IP hosts with router addresses by using DHCP. However, DHCP is quite a large protocol, and it was (wisely) determined that Mobile IP should not rely on the existence of another more compli-

cated protocol, especially one not widely available at the time the working group started.

It should be noted that the router discovery messages do not constitute a routing protocol. They enable hosts to discover the existence of neighboring routers, but not which router is best used to reach a particular destination. If a host chooses a poor first-hop router for a particular destination, it should receive an ICMP redirect from that router, which identifies a better one.

3.2.1 Router Discovery ICMP Message

Typically, a router implementing RFC 1256 will periodically multicast or broadcast a router advertisement to those links to which it is connected and to which it wishes to offer routing services. Then, each host equipped to understand the protocol will listen for the advertisements and be able to select a router address (typically only one is necessary) to use as a default router. The speed with which such hosts can choose a default router is then determined by the advertisement period. If a host fails to detect several consecutive Router Advertisements, the host can infer that the router is no longer offering service and can try to obtain service from a new router by listening for new advertisements.

One feature of these Router Advertisements is that the routers are allowed to denote, by setting *preference levels*, how eager they are to have new hosts using their services. A route that is advertised with a high preference level should be selected instead of another route that is advertised with a low preference level.

The message format for the Router Advertisement message is shown in Figure 3.1. The fields have the following meanings:

Type	9
Code	0
Checksum	the 16-bit one's complement of the one's complement sum of the ICMP message, starting with the ICMP type. To compute the checksum, the checksum field is set to 0.
Num addrs	the number of router addresses advertised in this message
Addr entry size	the number of 32-bit words of information for each router address (2, in the version of the protocol described here)
Lifetime	the maximum number of seconds that the router addresses may be considered valid
Router Address(i)	i = 1...num addrs, the sending router's IP addresses on the interface from which this message is sent

Preference level(i) i = 1...Num Addrs, the preferability of each corresponding router address as a default router address relative to other router addresses on the same subnet. The value is a signed, two's complement value; higher values are more preferable.

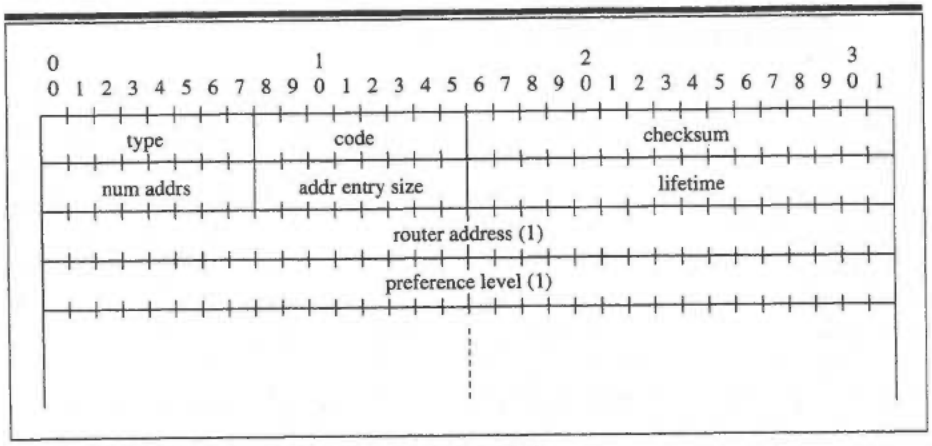

Figure 3.1 Router Advertisements (from RFC 1256).

Since this is an ICMP message, it is preceded by an IP header. In the IP header, the fields are set to mostly natural values. If the destination address is chosen to be the multicast address 224.0.0.1 (the all-systems multicast address), then the *TTL* (time to live) field is required to be set to 1.

3.2.2 Router Solicitation ICMP Message

When an IP host needs timely information about local default routers, it can multicast or broadcast a *router solicitation* message. Any routers in the vicinity that obey the router discovery protocol will respond with a unicast router advertisement message sent directly to the soliciting host. After receiving the advertisement, the host then responds just as if the advertisement were unsolicited and received at the broadcast or multicast address.

In Figure 3.2, the fields have the following meanings:

Type 10

Code 0

Checksum The 16-bit one's complement of the one's complement sum of the ICMP message, starting with the ICMP type. To compute the checksum, the checksum field is set to 0.

Reserved Sent as 0; ignored on reception.

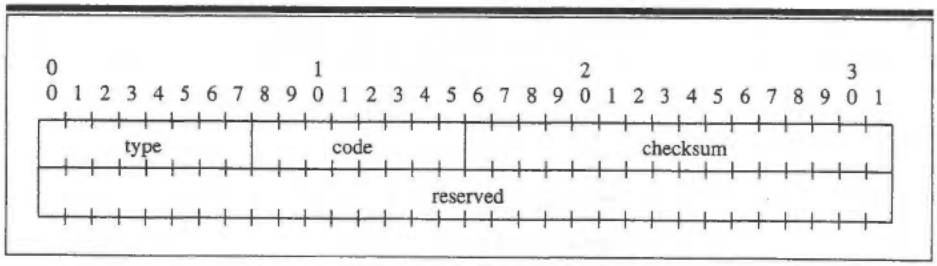

Figure 3.2 Router solicitations (from RFC 1256).

A host sending a solicitation is required to set the TTL field to 1. The only permissible values for the IP destination are the all-routers multicast address, 224.0.0.2, or the limited-broadcast address, 255.255.255.255.

3.3 Agent Advertisement

An agent advertisement is an ICMP Router Advertisement (as described in Section 3.2.2) that has been extended also to carry mobility agent advertisement extension (Section 3.3.1). A mobility agent transmits agent advertisements to advertise its services on a link. Mobile nodes use these advertisements to determine their current point of attachment to the Internet. The advertisement may also carry other extensions, notably the prefix-length extension (Section 3.3.2), one-byte padding extension (Section 3.3.3), or other extensions that might be defined in the future. Unquestionably the most important extension is the mobility agent extension. Within an agent advertisement message, ICMP Router Advertisements include the following link-layer, IP, and ICMP header fields:

- **Link-layer fields**

  Destination address — The link-layer destination address of a unicast agent advertisement is required to be the same as the source link-layer address of the agent solicitation that prompted the advertisement.

- **IP fields**

  TTL — The TTL for all agent advertisements is required to be set to 1.

Destination address — As specified for ICMP router discovery the IP destination address of an agent advertisement is required to be either the all-systems-on-this-link multicast address (224.0.0.1)(Deering 1989) or the limited-broadcast address (255.255.255.255). The subnet-directed broadcast address of the form <prefix>.<-1> cannot be used because mobile nodes will not generally know the prefix of the foreign network.

- **ICMP Fields**

Type — 9

Code — The code field of the agent advertisement is interpreted as follows:

- 0 — The mobility agent handles common traffic; that is, it acts as a router for IP datagrams not necessarily related to mobile nodes.
- 16 — The mobility agent does not route common traffic. However, all foreign agents are required to (at least) forward (possibly to their default router) any datagrams received from a registered mobile node (Section 5.9.2).

Lifetime — The lifetime is the maximum length of time that the advertisement is considered valid in the absence of further advertisements.

Router addresses — The usual router addresses present in any Router Advertisement may also appear in this portion of the agent advertisement (but see Section 3.5.1).

Num addrs — Num addrs is the number of router addresses advertised in the message.

Note that in an agent advertisement message, the number of router addresses specified in the ICMP Router Advertisement portion of the message may be set to zero. See Section 3.5.1 for details.

If sent periodically, the nominal interval at which agent advertisements are sent should be one third of the advertisement lifetime given in the ICMP header. This allows a mobile node to miss three successive advertisements before deleting the agent from its list of valid agents. The actual transmission time for each advertisement should be slightly randomized (Deering 1991) to avoid synchronization and subsequent collisions with agent advertisements sent by other agents, or with

Router Advertisements sent by other routers. Note that this field has no relation to the registration lifetime field within the mobility agent advertisement extension defined in the next section.

3.3.1 Mobility Agent Advertisement Extension

The mobility agent advertisement extension, illustrated in Figure 3.3, follows the ICMP Router Advertisement fields. It indicates that an ICMP router advertisement message is actually an agent advertisement being sent by a mobility agent.

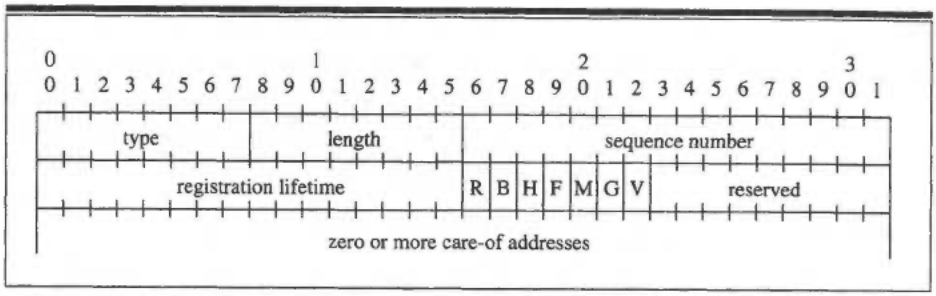

Figure 3.3 Mobility agent advertisement extension.

The individual fields of the mobility agent advertisement extension are defined as follows, with the bit fields denoted by their single-letter name:

Type	16
Length	(6 + 4*N), where N is the number of care-of addresses advertised
Sequence number	The count of agent advertisement messages sent since the agent was initialized (Section 3.5.2)
Registration lifetime	The longest lifetime (measured in seconds) that this agent is willing to accept in any registration request; A value of 65,535 indicates infinity.
R	Registration required. Registration with this foreign agent (or another foreign agent on this link) is required rather than using a colocated care-of address.
B	Busy. If this bit is set, the foreign agent will not accept registrations from additional mobile nodes.
H	Home agent. If this bit is set, this agent offers service as a home agent on the link on which the agent advertisement message is sent.

F	Foreign agent. This agent offers service as a foreign agent on the link on which the agent advertisement message is sent.
M	Minimal encapsulation. This agent implements receiving tunneled datagrams that use minimal encapsulation (Section 5.3).
G	Generic Routing Encapsulation (GRE). This agent implements receiving tunneled datagrams that use GRE (Section 5.4).
V	Van Jacobson header compression. This agent supports use of Van Jacobson header compression (Jacobson 1990) over this link with any registered mobile node.
Reserved	Sent as 0; ignored on reception
Care-of addresses	The advertised foreign agent care-of addresses provided by this foreign agent. An agent advertisement is required to include at least one care-of address if the F bit is set. The number of care-of addresses present is determined by the length of the extension.

A home agent must be prepared to serve its mobile nodes. In other words, the home agent should never claim to be too busy to serve the mobile nodes on its home network. To avoid overload, it is possible to configure mobile nodes and home agents so that there are multiple home agents on a home network, and so that the mobile nodes are divided into disjointed populations that report to the different home agents. Even in this case, however, an advertisement from any of the home agents on the same home network will suffice to inform the mobile node that it is indeed attached to its home network.

A foreign agent may at times be too busy to serve additional mobile nodes; even so, it must continue to send agent advertisements so that any mobile nodes already registered with it will know that they have not moved out of range of the foreign agent and that the foreign agent has not failed. A foreign agent may indicate that it is too busy to allow new mobile nodes to register with it, by setting the B bit in its agent advertisements. An agent advertisement message is not allowed to have the B bit set if the F bit is not also set. Either the F bit or the H bit is required to be set in the mobility agent advertisement extension.

When a foreign agent wishes to require registration even from those mobile nodes that have acquired a colocated care-of address, it sets the R bit to one. Because this

bit applies only to foreign agents, an agent is not allowed to set the R bit to 1 unless the F bit is also set to 1.

Note that the registration lifetime field has no relation to the advertisement lifetime field within the ICMP router advertisement portion of the agent advertisement. The latter field specifies the length of time before which the receiving node should consider that advertisements have been lost. Note also that the maximum registration lifetime permitted by the packet format is 65,534 seconds, which is slightly more than 18 hours.

3.3.2 Prefix-length Extension

The prefix-length extension may follow the mobility agent advertisement extension. It is used to indicate the number of bits of network prefix that apply to each router address listed in the ICMP Router Advertisement portion of the agent advertisement (Figure 3.1). Note that the prefix lengths given do not apply to the care-of addresses listed in the mobility agent advertisement extension.

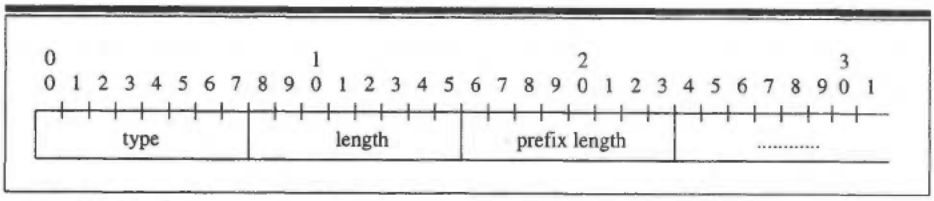

Figure 3.4 Prefix-length extension format.

The prefix-length extension is defined as shown in Figure 3.4, where

Type	19
Length	N, where N is the value of the num addrs field in the ICMP Router Advertisement portion of the agent advertisement
Prefix length	the number of leading bits that define the network number of the corresponding router address listed in the ICMP router advertisement portion of the message

The prefix length for each router address is encoded as a separate byte in the same order that the router addresses are listed. See Section 7.5.2 for information about how the prefix-length extension may be used by a mobile node when determining whether it has moved. There are some important implementation details that

must be kept in mind when using this extension; these are also detailed in Section 7.5.2.

3.3.3 One-byte Padding Extension

Some IP protocol implementations insist on padding ICMP messages to an even number of bytes. If the ICMP length of an agent advertisement is odd, this extension may be included to make the ICMP length even. Note that this extension is not intended to be a general-purpose extension to be included to word align or long align the various fields of the agent advertisement. An agent advertisement should not include more than one one-byte padding extension, and if present this extension should be the last extension in the agent advertisement.

Note that unlike other extensions used in Mobile IP, the one-byte padding extension is encoded as a single byte, with no length or data field present. The one-byte padding extension is defined in Figure 3.5, where type is set to 0 to denote one-byte padding extension.

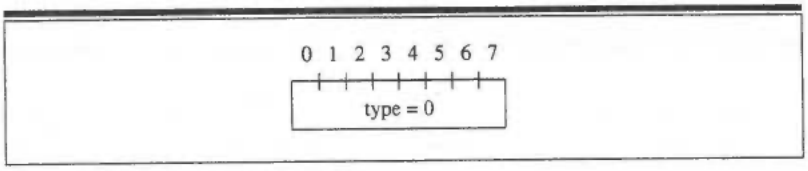

Figure 3.5 Pad extension format.

3.4 Agent Solicitation

The format of an agent solicitation is the same as an ICMP router solicitation, as shown previously in Figure 3.2. However, the way in which it is used is slightly different. For one thing, any agent solicitation used with Mobile IP is required to have the TTL field set to 1. There are other operational differences; see Section 3.6 for more details.

3.5 Mobility Agent Operation

Any mobility agent (home agent or foreign agent) that cannot be discovered by a link-layer protocol is required to implement agent advertisements. An agent that can be discovered by a link-layer protocol should also implement agent advertisements so that it can respond to agent solicitations. However, the advertisements need only be sent when the site policy requires registration with the agent (that is, when the R bit is set) or as a response to a specific agent solicitation.

The same procedures, defaults, and constants are used in agent advertisement messages as specified for ICMP router discovery (Section 3.2), except for the following.

- A mobility agent is required to limit the rate at which it sends broadcast or multicast agent advertisements. A recommended maximal rate is once per second.
- A foreign agent must accept router solicitations even when the IP source address appears to reside on a different subnet than the mobility agent's interface on which the solicitation was received.
- A mobility agent may be configured to send agent advertisements only in response to an agent solicitation message.

Refer again to Figure 2.3. Since the home address owned by a mobile node is typically not able to be located on any network attached to a foreign agent, the solicitation mechanism could not possibly work if the foreign agent disallowed solicitations from an apparently off-link IP address.

If the home network is not a *virtual network*, then the home agent for any mobile node should be located on the link identified by the mobile node's home address, and agent advertisement messages sent by the home agent on this link are required to have the H bit set. In this way, mobile nodes on their own home network are able to determine that they are indeed at home. If the home agent is attached to multiple links, it transmits agent advertisements with the H bit set only on those links for which it is willing to serve as a home agent.

If the home network is a virtual network, then it has no physical realization external to the home agent itself. In this case there is no physical network link on which to send agent advertisement messages advertising the home agent. Mobile nodes on a virtual home network are always treated as being away from home.

On a particular subnet, either all mobility agents are required to include the prefix-length extension, or none of them are allowed to include this extension. Equivalently, if any mobility agents on a given subnet include the extension, then all of them are required to include it. Otherwise, one of the move detection algorithms designed for mobile nodes will not function properly (see Section 7.5.2).

3.5.1 Advertised Router Addresses

The ICMP Router Advertisement portion of the agent advertisement may contain one or more router addresses. Thus, an agent may include one of its own addresses in the advertisement. A foreign agent may discourage use of this address as a default router by setting the preference to a low value and by including the address of another router in the advertisement (with a correspondingly higher preference). Nevertheless, a foreign agent is required, in every circumstance, to be able to route

datagrams it receives from registered mobile nodes (Section 5.9.2). Note that the mobile node is disallowed from broadcasting ARP packets on foreign networks; this is explained in Chapter 5, but the basic reason is to prevent the creation of ARP cache entries within other nodes on the foreign networks. ARP cache entries in other nodes will be incorrect after the mobile node moves again, and no protocol is established for the correction of such stale ARP cache entries. Without ARP it is difficult for the mobile node to discover link-layer addresses for the other routers in the advertisement, so the use of this feature is questionable at the present time.

3.5.2 Sequence Numbers and Rollover Handling

The sequence number in agent advertisements ranges from 0 to 65,535. After booting, an agent is required to use the number 0 for its first advertisement. Each subsequent advertisement is required to use the sequence number one greater, with the exception that the sequence number 65,535 is required to be followed by sequence number 256. In this way, mobile nodes can distinguish reductions in sequence numbers that result from reboots, from reductions that result in rollover of the sequence number after it attains the value 65,535. Since the mobile node can tell the difference, it does not have to register again with its home agent just because the sequence number from the foreign agent has rolled over. However, if the foreign agent reboots and thus reinitializes its sequence numbers starting from 0, then obviously the mobile node should reregister so that the foreign agent can be notified again of the mobile node's presence. It is expected that mobile nodes would never accidentally fail to detect 255 consecutive advertisements.

3.6 Agent Discovery by Mobile Nodes

Every mobile node is required to implement agent solicitations. Solicitations should only be sent in the absence of agent advertisements and when a care-of address has not been determined through a link-layer protocol or other means. The mobile node uses the same procedures, defaults, and constants for agent solicitation as specified for ICMP router solicitation messages, except that (1) the mobile node may solicit more often than once every three seconds and (2) a mobile node that is currently not connected to any foreign agent may solicit more times than MAX_SOLICITATIONS (as defined in RFC 1256). In fact, a mobile node can continue to send out solicitations indefinitely until a suitable foreign agent finally comes within range.

A mobile node is required to limit the rate at which it sends solicitations. The node may send three initial solicitations (on a given link) at a maximum rate of one per second while searching for an agent. After this, the solicitation rate is required

to be reduced so as to limit the overhead on the local link. Subsequent solicitations are required to be sent using a binary exponential backoff mechanism, doubling the interval between consecutive solicitations up to a maximal interval. The maximal interval should be chosen appropriately based on the characteristics of the media over which the mobile node is soliciting. According to the base mobile IP specification, this maximal interval should be at least one minute between solicitations, but it seems likely that for many uses this is too infrequent.

While still searching for an agent, the mobile node is not allowed to increase the rate at which it sends solicitations unless it has received a positive indication that it has moved to a new link. After successfully registering with an agent, the mobile node should also increase the rate at which it will send solicitations when it next begins searching for a new agent with which to register. The increased solicitation rate may revert to the maximal rate, but then is required to be limited in the manner described in the previous paragraph. In all cases the recommended solicitation intervals are nominal values. Mobile nodes are expected to randomize their solicitation times around these nominal values as specified for ICMP router discovery.

Mobile nodes process agent advertisements to discover a care-of address (and a foreign agent)—a parameter crucial to the successful operation of Mobile IP. A mobile node can distinguish an agent advertisement message from other uses of the ICMP Router Advertisement message by examining the number of advertised addresses and the IP total length field. When the IP total length indicates that the ICMP message is longer than needed for the number of advertised addresses, the remaining data is interpreted as one or more extensions. The presence of a mobility agent advertisement extension naturally identifies the advertisement as an agent advertisement.

When multiple methods of agent discovery are in use, the mobile node should first attempt registration with agents that include mobility agent advertisement extensions in their advertisements, in preference to those discovered by other means. This preference maximizes the likelihood that the registration will be recognized, thereby minimizing the number of registration attempts. Otherwise it might be possible, for instance, to attempt registration with a wireless access point that was not offering any care-of address.

3.6.1 Registration Required

When the mobile node receives an agent advertisement with the R bit set, the mobile node should register through the foreign agent, even when the mobile node might be able to acquire its own colocated care-of address. This feature is intended to allow sites to enforce visiting policies (such as accounting), which require exchanges of authorization. The intention is to simplify matters for mobile nodes in such domains, and to eliminate one possible cause for rejection and delay.

3.6.2 Returning Home

A mobile node can detect that it has returned to its home network when it receives an agent advertisement from its own home agent. If so, it should deregister with its home agent (Section 4.3). Before attempting to deregister, the mobile node should configure its routing table appropriately for its home network (Section 5.9.1). In addition, if the home network is using ARP (Plummer 1982), the mobile node is required to follow the procedures described in Chapter 5 with regard to ARP, proxy ARP, and gratuitous ARP.

3.7 Second Thoughts on Using RFC 1256

As stated, the original motivation for using the Router Advertisement protocol with Mobile IP was to simplify development. Router Advertisement was not originally designed to handle mobile nodes, but it seemed like such a natural fit, given that the purpose was to provide a means by which a mobile computer could discover a foreign agent (called for the purposes of this discussion, its *default router*). However, the attempted reuse has had the opposite effect. In the first place, router discovery defines a collection of *configuration variables*, which any implementation has to set and use correctly. Unfortunately, there have been frequent technical debates on whether the configuration variables appropriate for general router discovery were also appropriate for discoverying mobility agents. For instance, consider the configuration variable MinAdvertisementInterval. In RFC 1256, this variable is required to be set to no less than three seconds. Such a value would render the advertisement feature almost useless for many wireless mobile nodes.

To see why, suppose a mobile node makes a cell switch. To detect the movement at the network layer (as detailed in Section 7.5), the mobile node has to hear a mobility agent advertisement from another foreign agent. If the mobile node has to wait three seconds to discover that its previous foreign agent is out of reach, an unacceptably jerky response time will be observed. Worse yet, a mobile node will not typically make a cell switch just because a single advertisement was lost. Often the foreign agent is still available but that the advertisement has experienced a collision during its transmission to the communications medium. Following the stated values in RFC 1256 would typically lead to waiting at least six seconds to determine that a cell switch should occur.

The result of obeying that requirement would be that all mobility agent discovery operations would proceed by way of using solicitations. This is an undesirable result, especially since each mobile node might issue a solicitation each second, consuming bandwidth unnecessarily.

Even with solicitations, however, there are other problems involved with following the dictates of RFC 1256. For instance, there is another protocol constant

(MAX_SOLICITATIONS) that limits how many times a soliciting IP host can request a Router Advertisement. In the wired world, this makes sense because there is little point to continuing to stimulate dead routers on a wire that could be supporting a great deal of routerless local traffic. Furthermore, if the solicitations are sent to the broadcast address 255.255.255.255, every other host on the network would in that case be interrupted to process a meaningless packet.

Contrast this with the case for wireless. A wireless mobile node, out of range of every foreign agent, is likely to issue solicitations indefinitely until a base station or other wireless access point becomes available. And it is less likely (although not completely unlikely) that this behavior will take away from the effectiveness of other IP hosts.

Within the Mobile IP Working Group, the issue of preferences for mobility agent discovery was among the most hotly debated topics. With the unmodified router discovery, a case (albeit weak) could be made for the use of preferences, which allow routers to be selected in a particular order. With Mobile IP a mobile node typically looks for exactly one foreign agent. Even though some commentators thought that a mobile node in the presence of multiple foreign agents should be able to select one with the highest preference, no one was ever able to describe just how the foreign agents would be able to adjust their preference levels dynamically. The ways that were suggested seemed unable to promote interoperability and avoid possible oscillatory behaviors. Moreover, even when preferences were in use, from the beginning it was prohibited for mobile nodes to move away from a foreign agent purely for the reason that it had begun to issue advertisements with lower preferences. Combined with various other perceived difficulties, these points eventually motivated the Working Group to eliminate the use of preferences entirely from the Mobile IP protocol. They have not been missed.

It might seem that foreign agents should be able to advertise the IP addresses of other routers that are attached to the same link as the mobile node. In fact, for these other routers (which are likely not to have care-of addresses) the foreign agent could, just as always, specify preferences so that the mobile node could make an informed selection. However, as detailed in Section 5.13, the mobile node is expressly forbidden to broadcast any ARP packets. Therefore, it is not clear how the mobile node could ever discover the link-layer address of any other router besides the foreign agent, and Mobile IP does not specify a method for doing so. In other words, currently it is useless for the foreign agent to advertise any other routers in its agent advertisement messages.

One additional inconvenience of using RFC 1256 has only recently surfaced. In some commercial TCP/IP product, there is no easy way for user (nonoperating system) code to issue or receive ICMP datagrams. In other words, such commercial products unfortunately do not offer a suitable API for ICMP. Since some Mobile IP products would typically be sold as nonoperating system applications to be added

after the initial purchase, this restriction is a problem that has a substantial effect on the design and implementation of such mobility software.

3.8 Summary

The ICMP Router Advertisement protocol is modified to enable mobile nodes to detect Mobile IP home agents and foreign agents. The model is appropriate on one level, since the functions of home agents and foreign agents can be carried out by specialized routers. Much of the mechanism defined for ICMP Router Advertisement is, however, unnecessary for Mobile IP, and vice versa. In particular, the preferences are not used for care-of addresses, and a mobile node is prevented from putting its home address in any ARP requests used to discover link-layer addresses for other default routers besides the foreign agent. Mobile nodes may solicit for service using roughly the same procedures as defined for ICMP Router Advertisement, except the procedures are allowed to be carried out more often as necessary.

The agent advertisement extension is the most important extension defined for Mobile IP, but there is also a prefix-length extension that is useful especially with mobile nodes connected to wired networks. Mobility agents use the agent advertisement extension to make themselves detectable to mobile nodes, and foreign agents include one or more care-of addresses in the advertisement. The availability of other services is indicated by bits in the extension header, including various encapsulations and Van Jacobson header compression. The agent advertisement is also used for other proposed extensions to the basic Mobile IP protocol, some of which are described in later chapters.

This chapter also described the relevant operational procedures and rules by which ICMP messages are to be used by mobile computers and mobility agents. This lays the informational groundwork for the Mobile IP registration procedure, which forms the subject of the next chapter.

CHAPTER FOUR

Registration

Mobile IP registration provides a flexible and reliable mechanism for mobile nodes to communicate their current reachability information to their home agent. It is the method by which mobile nodes

- Request forwarding services when visiting a foreign network
- Inform their home agent of their current care-of address
- Renew a binding that is due to expire
- Deregister when they return home

Registration messages exchange the mobile node's current binding information among a mobile node, its home agent, and (possibly) a foreign agent. Registration creates or modifies a mobility binding at the home agent, associating the mobile node's home address with its care-of address for a certain length of time, called the *registration lifetime* (or usually just *lifetime* when there is no chance for confusion with the lifetime associated with the periodic arrival of an agent advertisement).

Several other optional capabilities are available through the registration procedure, which enables a mobile node to

- Discover the address of a home agent if the mobile node is not configured with this information
- Select certain alternative tunneling protocols (minimal encapsulation or GRE)
- Request the use of Van Jacobson (Jacobson 1990) header compression
- Maintain multiple simultaneous registrations so that a copy of each datagram will be tunneled to each active care-of address
- Deregister certain care-of addresses while retaining others

4.1 Registration Overview

A mobile node is required to be configured with its home address and a netmask (as described in Section 1.5.1), and a mobility security association for each home agent. In addition, a mobile node may be configured with the IP address of one or more of its home agents; otherwise, the mobile node may discover a home agent using the procedures described in Section 4.6.3.

Mobile IP has two variations of its registration procedures—one by means of a foreign agent that relays the registration to the mobile node's home agent and one without any such intermediary. The following rules determine which of these two registration procedures to use in any particular circumstance.

- If a mobile node is registering a foreign agent care-of address, the mobile node is required to register via that foreign agent.
- Under any circumstances, if a mobile node receives an agent advertisement from a foreign agent with the R bit set, the mobile node should register via a foreign agent.
- If a mobile node has returned to its home network and is deregistering with its home agent, the mobile node sends the registration addressed directly to its home agent.
- Likewise, if a mobile node is using a colocated care-of address, the mobile node naturally sends the registration addressed directly to its home agent.

Both registration procedures involve the exchange of registration request and registration reply messages (Sections 4.3 and 4.4). When registering by way of a foreign agent, the registration procedure requires the following four messages, as illustrated in Figure 4.1.

1. The mobile node sends a registration request to the prospective foreign agent to begin the registration process.
2. The foreign agent processes the registration request and then relays it to the home agent, whose address is provided by the mobile node in the registration request.
3. The home agent sends a registration reply to the foreign agent to grant or deny the request.
4. The foreign agent processes the registration reply and then relays it to the mobile node to inform it of the disposition of its request.

When the mobile node registers directly with its home agent, the registration procedure requires only the following two messages.

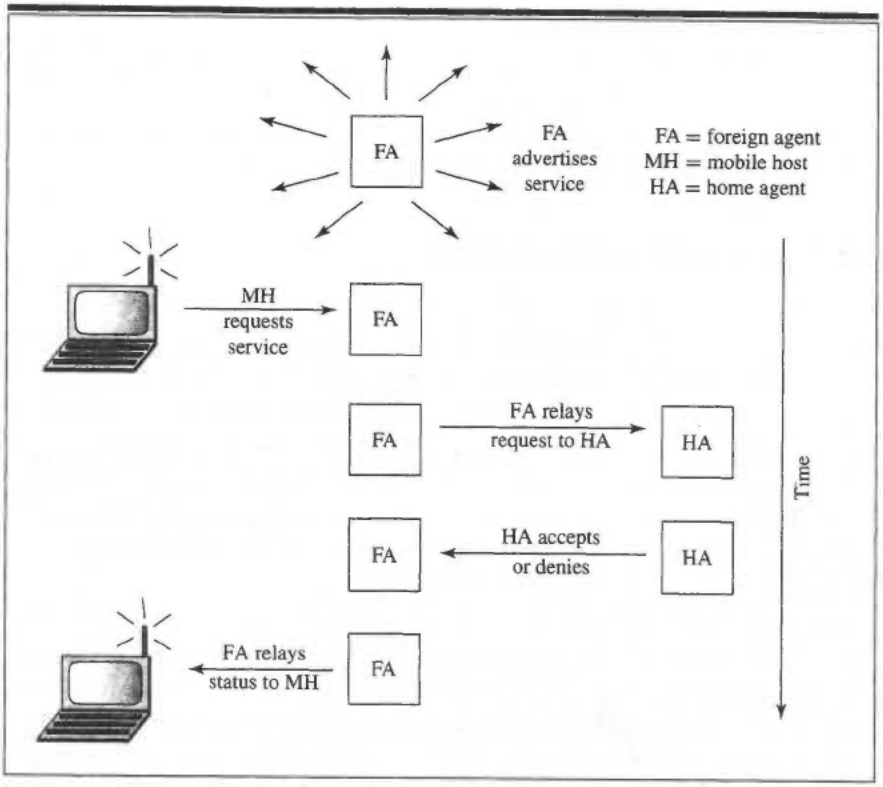

Figure 4.1 Mobile IP registration overview.

1. The mobile node sends a registration request to the home agent.
2. The home agent sends a registration reply to the mobile node that grants or denies the request.

Mobile IP registration messages use the User Datagram Protocol (UDP) (Postel 1980). The overall data structure of the registration messages is shown in Figure 4.2. A nonzero UDP checksum should be included in the header, and is then required to be checked by the recipient. UDP is specified instead of TCP for transporting registration messages, because Mobile IP does not need the windowing, renumbering, congestion control, or flow control that TCP provides. Mobile IP defines its own retransmissions to handle cases of dropped packets. Moreover, and especially in the case of wireless communications, TCP can perform poorly when packets are dropped because of noisy or lossy channels.

Registration messages contain a lifetime field that indicates the amount of time (in seconds) for which the registration information should be considered valid. A value of 0 indicates that the mobile node has been deregistered. A value of 65,535 indicates infinity.

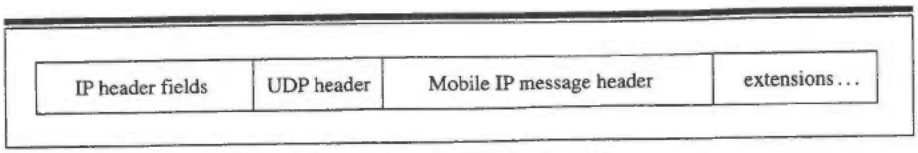

Figure 4.2 General Mobile IP registration message format.

4.2 Authentication Overview

Each mobile node, foreign agent, and home agent is required to be able to support a *mobility security association* for mobile entities, indexed by their *security parameters index* (*SPI*) and IP address. In the case of the mobile node, the latter must be its home address. Section 4.9.1 discusses requirements for supporting authentication algorithms. Registration messages between a mobile node and its home agent are required to be authenticated with the mobile-home authentication extension (Section 4.5.1). This extension immediately follows all nonauthentication extensions, except those foreign agent-specific extensions that may be added to the message after the mobile node computes the authentication.

If a malicious agent were able to snoop on a mobile node during its registration process, it could collect all the necessary data for that registration, including the necessary authentication data. This registration data could be replayed at some future date, and since the authentication was computed by the mobile node it would still be valid. Thus, something has to change to make the registration data different each time; this change is found in the identification field of the registration request. Replay protection in Mobile IP is accomplished by using a different (fresh) value in the identification field of each registration message.

The registration reply also contains an identification value, and it is based on the identification field from the registration request message from the mobile node. The reply identification also depends on the style of replay protection used between the mobile node and its home agent. Each such security selection is associated with one entry of the mobility security association between the mobile node and the home agent. The particular selection is indicated by the SPI value in the mobile-home authentication extension. Note that SPI values 0 through 255 are reserved and cannot be used in any mobility security association. The authentication procedures are fully described in Sections 4.9.4 and 4.9.6.

4.3 Registration Request

A mobile node registers with its home agent using a registration request message so that its home agent can create or modify a *mobility binding* for that mobile node (for example, additional lifetime). The request may be relayed to the home agent by the foreign agent through which the mobile node is registering or it may be sent

directly to the home agent when the mobile node is registering a colocated care-of address. Fields in the various headers of the request message are set as listed.

- **IP fields**

Source address	Typically the interface address from which the message is sent
Destination address	Typically that of the foreign agent or the home agent

- **UDP fields**

Source Port	Variable
Destination Port	434

- **Mobile IP fields**

The UDP header is followed by the Mobile IP fields shown in Figure 4.3, with the fields defined as follows.

Type	1 (registration request)
S	Simultaneous bindings. By setting the S bit, the mobile node is requesting that the home agent retain its prior mobility bindings.
B	Broadcast datagrams. By setting the B bit, the mobile node requests that the home agent tunnel to it any broadcast datagrams that it receives on the home network, as described in Section 5.10. See also Section 7.3.1 for some more recent efforts.
D	Decapsulation. By setting the D bit, the mobile node informs the home agent that it will decapsulate datagrams that are sent to the care-of address. That is, the mobile node is using a colocated care-of address.
M	Minimal encapsulation. By setting the M bit, the mobile node requests that its home agent use minimal encapsulation (Perkins 1996c) for datagrams tunneled to the mobile node.
G	GRE encapsulation. By setting the G bit, the mobile node requests that its home agent use GRE encapsulation (Hanks et al. 1994a) for datagrams tunneled to the mobile node.
V	Van Jacobson header compression. By setting the 'V' bit, the mobile node requests that its mobility agent use Van Jacobson header compression (Jacobson 1990) over its link with the mobile node.
rsv	Reserved bits; sent as 0, ignored on reception

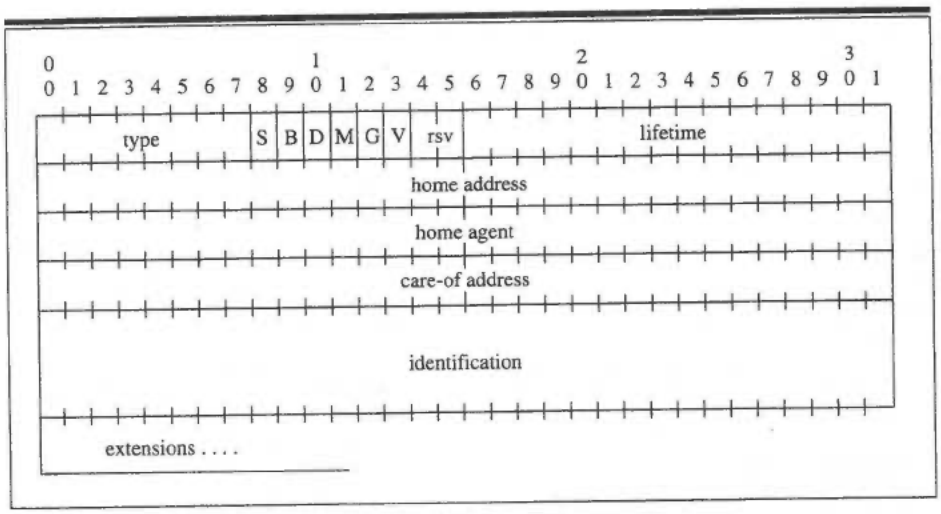

Figure 4.3 Registration request packet format.

Lifetime	The number of seconds remaining before the registration is considered expired
Home address	The IP address of the mobile node
Home agent	The IP address of the mobile node's home agent
Care-of address	The IP address for the tunnel endpoint
Identification	A 64-bit number constructed by the mobile node and used for matching registration requests with registration replies, as well as for protecting against replay attacks of registration messages. (Sections 4.9.4 and 4.9.6)
Extensions	What follows the fixed portion of the registration request

See Sections 4.6.1 and 4.7.2 for information on the relative order in which different extensions, when present, are required to be placed in a registration request message.

4.4 Registration Reply

As described, mobility agents return a registration reply message to a mobile node that has sent a registration request message. If the mobile node is requesting service from a foreign agent, that foreign agent will receive the reply from the home agent and subsequently relay it to the mobile node. If, on the other hand, a mobile node has a colocated care-of address, it will receive the registration reply from its home agent. The reply message informs the mobile node of the status of its request and

indicates the lifetime granted by the home agent, which may be smaller than the original request.

The foreign agent is not allowed to modify the lifetime selected by the mobile node in the registration request, because the lifetime is covered by the mobile-home authentication extension, which cannot be correctly computed by the foreign agent. The home agent is not allowed to increase the lifetime selected by the mobile node in the registration request, because doing so could increase it beyond the maximal registration lifetime allowed by the foreign agent. If the lifetime received in the registration reply is greater than that in the registration request, the lifetime in the request is required to be used. When the lifetime received in the registration reply is less than that in the registration request, the lifetime in the reply is required to be used.

The following lists present the fields in the IP header, the fields in the UDP header, and the fields in the registration request message itself.

- **IP fields**

Source address	Typically copied from the destination address of the registration request to which the agent is replying. (See Sections 4.7.3 and 4.8.3 for details.)
Destination address	Copied from the source address of the registration request to which the agent is replying

- **UDP fields**

Source port	variable
Destination port	Copied from the source port of the corresponding registration request (Section 4.7.1)

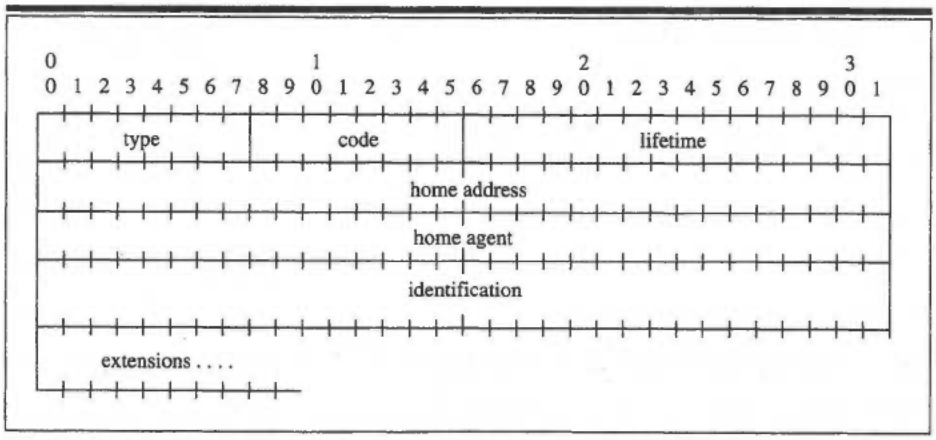

Figure 4.4 Registration reply packet format.