

implemented multicast routing protocol is the Multicast Open Shortest Path First protocol (MOSPF) [RFC 1584]. MOSPF operates in an autonomous system (AS) that uses the OSPF protocol (see Section 4.5) for unicast routing. MOSPF extends OSPF by having routers add their multicast group membership to the link state advertisements that are broadcast by routers as part of the OSPF protocol.

Inter-Autonomous System Multicast Routing

In our discussion above, we have implicitly assumed that all routers are running the same multicast routing protocol. As we saw with unicasting, this will typically be the case within a single autonomous system (AS). However, different ASs may choose to run different multicast routing protocols. Interoperability rules have been defined for the major Internet multicast routing protocols [RFC 2715]. (The rules are particularly messy due to the very different approaches taken to multicast routing by sparse- and dense-mode protocols.) What is still missing, however, is an *inter-AS* multicast routing protocol to route multicast datagrams among different ASs.

DVMRP has been the *de facto* inter-AS multicast routing protocol. However, as a dense-mode protocol, it is not particularly well suited to the rather sparse set of routers participating in today's Internet Mbone. The development of an inter-AS multicast protocol is an active area of research and development being carried out by the *idmr* working group of the IETF [IDMR 2002]. For a discussion of inter-AS multicast routing, as well as commentary on the multicast service model, deployment, and future, the reader is encouraged to consult [Diot 2000].

4.9 ♦ Mobility and the Network Layer

People are increasingly on the move today, and, with the proliferation of laptops, palmtops, PDAs, and mobile phones, this means that an increasing number of networked devices are also on the move. Our discussion of the network layer thus far has implicitly assumed a static network infrastructure, with a host having the same point of attachment (that is, first-hop router and edge network) into the larger network over time. In this section, we'll consider a mobile node that can change its point of attachment into the network over time. We'll see that mobility will require a number of important additions to the network-layer architecture.

4.9.1 Mobility Considerations in Network-Layer Design

Since the term **mobility** has taken on many meanings in both the computer and telephony worlds, it will serve us well first to consider several dimensions of mobility in some detail.

◆ **From the network layer's standpoint, how mobile is a user?** A physically mobile user will present a very different set of challenges to the network layer, depending on how he or she moves between points of attachment to the network. At one end of the spectrum, a user may carry a laptop with a wireless network interface card around in a building. This user is *not* mobile from a network-layer perspective if the same wireless link is used, regardless of location. At the other end of the spectrum, consider the user zooming along the autobahn in a BMW at 150 kilometers per hour, passing through multiple wireless access networks and wanting to maintain an uninterrupted connection to a remote application throughout the trip. This user is *definitely* mobile! In between these extremes is a user who takes a laptop from one location (for example, office or dormitory) into another (for example, coffeeshop, classroom, house, or other office building) and wants to connect into the network in the new location. This user is also mobile (although less so than the BMW driver!) but does not need to maintain an ongoing connection while moving between points of attachment to the network. Figure 4.58 illustrates this spectrum of user mobility from the network layer's perspective.

◆ **How important is the mobile node's address?** With mobile telephony, your phone number—essentially the network-layer address of your phone—remains the same as you travel from one provider's mobile phone network to another. Must a laptop similarly maintain the same IP address while moving between IP networks?

The answer to this question will depend strongly on the applications being run. For the BMW driver who wants to maintain an uninterrupted TCP connection to a remote application while zipping along the autobahn, it would be convenient to maintain the same IP address. Recall from the preceding chapter that an Internet application needs to know the IP address and port number of the remote entity with which it is communicating. If a mobile entity is able to maintain its IP address as it moves, mobility becomes invisible from the application standpoint. There is great value to this transparency—an application need not be concerned with a potentially changing IP address, and the same application code serves mobile and nonmobile connections alike. We will see in Section 4.9.3 that

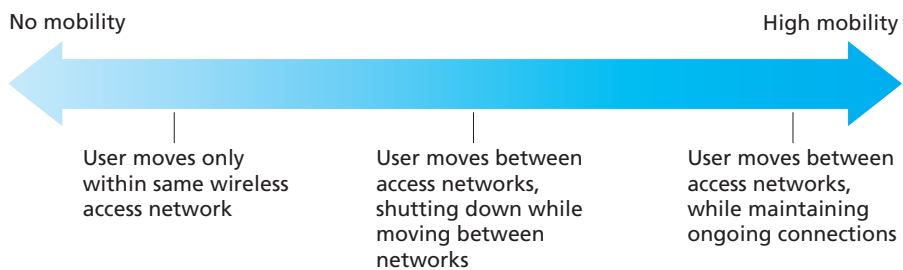


Figure 4.58 ◆ Various degrees of mobility, from the network layer's point of view

so-called “mobile IP” provides this transparency, allowing a mobile node to maintain its permanent IP address while moving among networks.

On the other hand, a less glamorous mobile user might simply want to turn off an office laptop, bring that laptop home, power up, and work from home. If the laptop will function primarily as a client in client-server applications (for example, send/read e-mail, browse the Web, Telnet to a remote host) from home, the particular IP address used by the laptop is not that important. In particular, one could get by fine with an address that is temporarily allocated to the laptop by the ISP serving the home. We saw in Section 4.4 that DHCP, the Dynamic Host Configuration Protocol, already provides this functionality.

- ◆ **What supporting wired infrastructure is available?** In all of our scenarios above, we’ve implicitly assumed that there is a fixed infrastructure to which the mobile user can connect, for example, the home’s ISP network, the wireless access network in the office, or the wireless access networks lining the autobahn. What if no such infrastructure exists? If two users are within communication proximity of each other, can they establish a network connection in the absence of any other network-layer infrastructure? So-called **ad hoc networking** provides precisely these capabilities. This rapidly developing area is at the cutting edge of mobile networking research and is beyond the scope of this book. [Perkins 2000] and the IETF Mobile Ad Hoc Network (manet) working group Web pages [manet 2002] provide thorough treatments of the subject.

As noted above, the dynamic address allocation techniques such as those used in DHCP (Section 4.4) allow for only a limited form of mobility. A mobile node can connect to the network at a chosen point of attachment but cannot run networked applications while moving between points of attachment. Let’s now consider the case where the node wants to maintain its address and continue its network connections uninterrupted as it moves between networks. We’ll first look at the principles and techniques that allow for such mobility and then consider the embodiment of these principles in the mobile IP standard.

4.9.2 Mobility Management

In order to illustrate the issues involved in allowing a mobile user to maintain ongoing connections while moving between networks, let’s consider a human analogy. A teenager or twenty-something adult moving out of the family home becomes mobile, living in a series of dormitories and/or apartments, and often changing addresses. If an old friend wants to get in touch, how can that friend find the address of his or her mobile friend? One common way is to contact the family, since a mobile youth will often register his or her current address with the family (if for no other reason than so that the parents can send money to help pay the rent!). The family home, with its permanent address, becomes that one place that others can go as a first step in communicating with the mobile youth. Later communication by the

friends may be either indirect (for example, with mail being sent first to the parents' home and then forwarded to the mobile youth) or direct (for example, with the friend using the obtained address to send mail directly to his or her mobile friend).

In a network setting, the permanent "home" of a mobile node (such as a laptop or PDA) is known as the **home network**, and the entity within the home network that performs the mobility management functions discussed below on behalf of the mobile node is known as the **home agent**. The network in which the mobile node is currently residing is known as the **foreign** (or **visited**) **network**, and the entity within the foreign network that helps the mobile node with the mobility management functions discussed below is known as a **foreign agent**. For mobile professionals, their home network might likely be their company network, while the visited network might be the network of a colleague they are visiting. A **correspondent** is the entity wishing to communicate with the mobile node. Figure 4.59 illustrates these concepts, as well as addressing concepts considered below. In Figure 4.59, we note that agents are shown as being collocated with routers (for example, as processes running on routers), but alternatively they could be executing on other hosts or servers in the network.

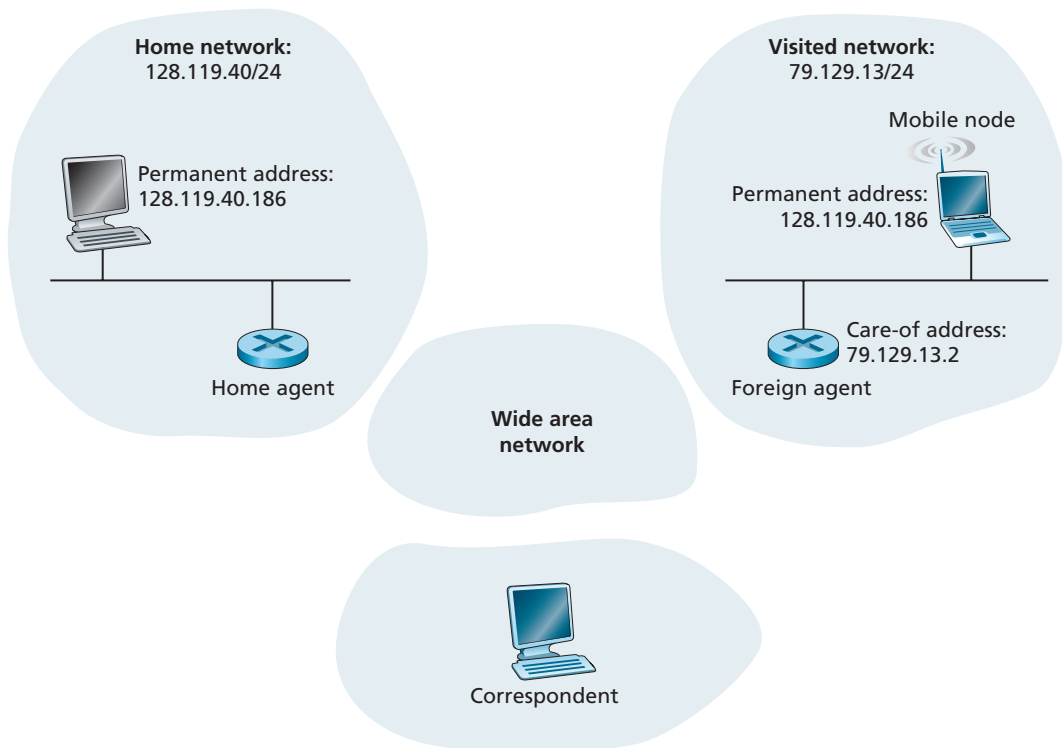


Figure 4.59 ♦ Initial elements of a mobile network architecture

Addressing

We noted above that in order for user mobility to be transparent to network applications, it is desirable for a mobile node to keep its address as it moves from one network to another. When a mobile node is resident in a foreign network, all traffic addressed to the node's permanent address now needs to be routed to the foreign network. How can this be done? One option is for the foreign network to advertise to all other networks that the mobile node is resident in its network. This could be via the usual exchange of intradomain and interdomain routing information and would require few changes to the existing routing infrastructure. The foreign network could simply advertise to its neighbors that it has a highly specific route to the mobile node's permanent address (that is, essentially inform other networks that it has the correct path for routing datagrams to the mobile node's permanent address; see Section 4.4.1). These neighbors would then propagate this routing information throughout the network as part of the normal procedure of updating routing information and forwarding tables. When the mobile node leaves one foreign network and joins another, the new foreign network would advertise a new, highly specific route to the mobile node, and the old foreign network would withdraw its routing information regarding the mobile node.

This solution solves two problems at once, and it does so without making significant changes to the network-layer infrastructure. Other networks know the location of the mobile node, and it is easy to route datagrams to the mobile node, since the forwarding tables will direct datagrams to the foreign network. A significant drawback, however, is that of scalability. If mobility management were to be the responsibility of network routers, the routers would have to maintain forwarding table entries for potentially millions of mobile nodes. Some additional drawbacks are explored in the problems at the end of this chapter.

An alternative approach (and one adopted in practice) is to push mobility functionality from the network core to the network edge—a recurring theme in our study of Internet architecture. A natural way to do this is via the mobile node's home network. In much the same way that parents of the mobile teenager track his or her location, the home agent in the mobile node's home network can track the foreign network in which the mobile node resides. A protocol between the mobile node (or a foreign agent representing the mobile node) and the home agent will certainly be needed to update the mobile node's location.

Let's now consider the foreign agent in more detail. The conceptually simplest approach, shown in Figure 4.59, is to locate foreign agents at the edge routers in the foreign network. One role of the foreign agent is to create a so-called **care-of address (COA)** for the mobile node, with the network portion of the COA matching that of the foreign network. There are thus two addresses associated with a mobile node, its **permanent address** (analogous to our mobile youth's family's home address) and its COA, sometimes known as a **foreign address** (analogous to the address of the house in which our mobile youth is currently residing). In the example in Figure 4.59, the permanent address of the mobile node is 128.119.40.186. When visiting network 79.129.13/24, the mobile node has a COA of 79.129.13.2. A second role of the

foreign agent is to inform the home agent that the mobile node is resident in its (the foreign agent's) network and has the given COA. We'll see shortly that the COA will be used to "reroute" datagrams to the mobile node via its foreign agent.

Although we have separated the functionality of the mobile node and the foreign agent, it is worth noting that the mobile node can also assume the responsibilities of the foreign agent. For example, the mobile node could obtain a COA in the foreign network (for example, using a protocol such as DHCP) and itself inform the home agent of its COA.

We have now seen how a mobile node obtains a COA, and how the home agent can be informed of that address. But having the home agent know the COA solves only part of the problem. How should datagrams be addressed and forwarded to the mobile node? Since only the home agent (and not network-wide routers) knows the location of the mobile node, it will no longer suffice to simply address a datagram to the mobile node's permanent address and send it into the network-layer infrastructure. Something more must be done. Two different approaches can be identified, which we will refer to as indirect and direct routing.

Indirect Routing to a Mobile Node

Let's first consider a correspondent that wants to send a datagram to a mobile node. In the **indirect routing** approach, the correspondent simply addresses the datagram to the mobile node's permanent address, and sends the datagram into the network, blissfully unaware of whether the mobile node is resident in its home network or is visiting a foreign network; mobility is thus completely transparent to the correspondent. Such datagrams are first routed, as usual, to the mobile node's home network. This is illustrated in Step 1 in Figure 4.60.

Let's now turn our attention to the home agent. In addition to being responsible for interacting with a foreign agent to track the mobile node's COA, the home agent has another very important function. Its second job is to be on the lookout for arriving datagrams addressed to nodes whose home network is that of the home agent, but that are currently resident in a foreign network. The home agent intercepts these datagrams and then "reroutes" them to a mobile node in a two-step process. The datagram is first forwarded to the foreign agent, using the mobile node's COA (Step 2 in Figure 4.60), and then forwarded from the foreign agent to the mobile node (Step 3 in Figure 4.60).

It is instructive to consider this rerouting in more detail. The home agent will need to address the datagram using the mobile node's COA, so that the network layer will route the datagram to the foreign network. On the other hand, it is desirable to leave the correspondent's datagram intact, since the application receiving the datagram should be unaware that the datagram was forwarded via the home agent. Both goals can be satisfied by having the home agent **encapsulate** the correspondent's original complete datagram within a new (larger) datagram. This larger datagram is addressed and delivered to the mobile node's COA. The foreign agent, who "owns" the COA, will receive and decapsulate the datagram, that is, remove the correspondent's original datagram from within the larger encapsulating datagram,

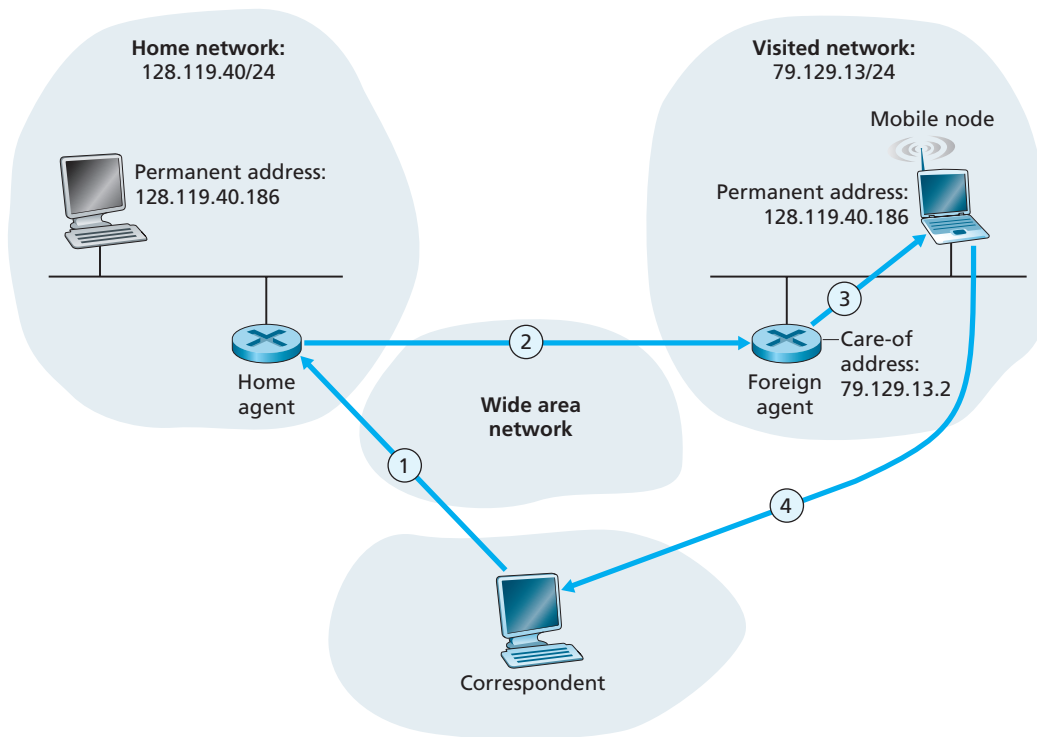


Figure 4.60 ♦ Indirect forwarding to a mobile node

and then forward (Step 3 in Figure 4.60) the original datagram to the mobile node. Figure 4.61 shows a correspondent's original datagram being sent to the home network, an encapsulated datagram being sent to the foreign agent, and the original datagram being delivered to the mobile node. The sharp reader will note that the encapsulation/decapsulation described here is identical to the notion of tunneling, discussed earlier in this chapter in the context of IP multicast and IPv6.

Let's next consider how a mobile node sends datagrams to a correspondent. This is quite simple, as the mobile node can address its datagram *directly* to the correspondent (using its own permanent address as the source address, and the correspondent's address as the destination address). Since the mobile node knows the correspondent's address, there is no need to route the datagram back through the home agent. This is shown as Step 4 in Figure 4.60.

Let's summarize our discussion of indirect routing by listing the new network-layer functionality required to support mobility.

- ♦ *A mobile node-to-foreign agent protocol:* for the mobile node to register with the foreign agent when attaching to the foreign network. Similarly, a mobile node will deregister with the foreign agent when it leaves the foreign network.

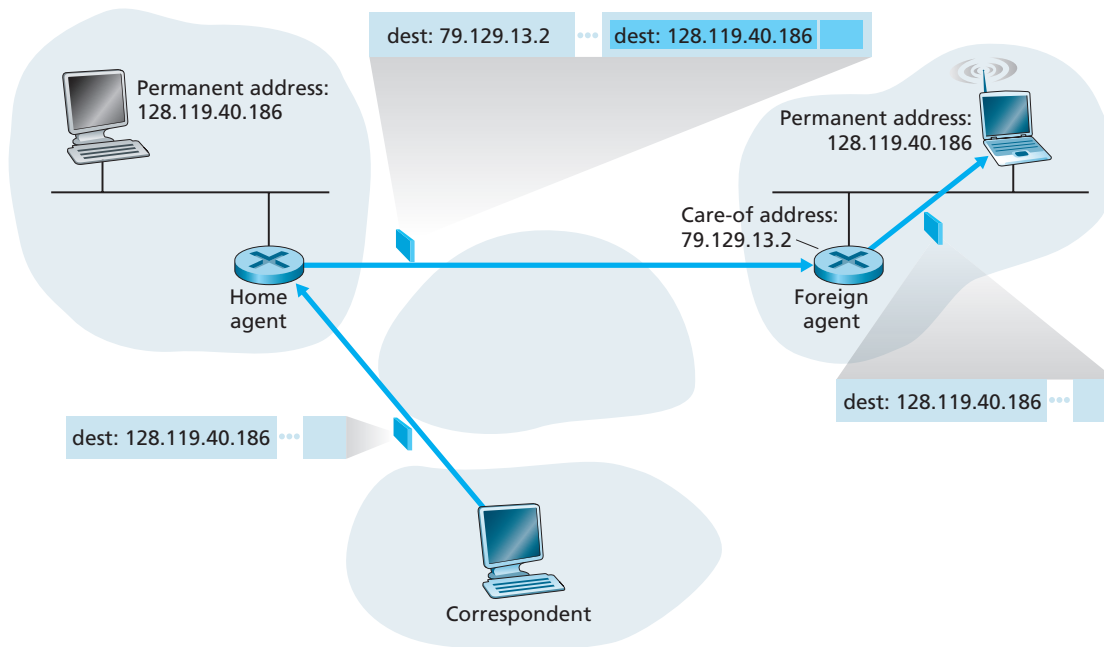


Figure 4.61 ♦ Encapsulation and decapsulation

- ♦ *A foreign agent-to-home agent registration protocol:* for the foreign agent to register the mobile node's COA with the home agent. A foreign agent need not explicitly deregister a COA when a mobile node leaves its network, as the subsequent registration of a new COA, when the mobile node moves to a new network, will take care of this.
- ♦ *A home-agent datagram encapsulation protocol:* encapsulation and forwarding of the correspondent's original datagram within a datagram addressed to the COA.
- ♦ *A foreign-agent decapsulation protocol:* extraction of the correspondent's original datagram from the encapsulating datagram, and the forwarding of the original datagram to the mobile node.

The discussion above provides all the pieces—foreign agents, the home agent, and indirect forwarding—needed for a mobile node to maintain an ongoing connection while moving among networks. As an example of how these pieces fit together, assume the mobile node is attached to foreign network *A*, has registered a COA in network *A* with its home agent, and is receiving datagrams that are being indirectly routed through its home agent. The mobile node now moves to foreign network *B* and registers with the foreign agent in network *B*, who informs the home agent of the mobile node's new COA. From this point on, the home agent will reroute datagrams to foreign network *B*. As far as a correspondent is concerned, mobility is

transparent—datagrams are routed via the same home agent both before and after the move. As far as the home agent is concerned, there is no disruption in the flow of datagrams—arriving datagrams are first forwarded to foreign network A; after the change in COA, datagrams are forwarded to foreign network B. But will the mobile node see an interrupted flow of datagrams as it moves between networks? As long as the time between the mobile node's disconnection from network A (at which point it can no longer receive datagrams via A) and its attachment to network B (at which point it will register a new COA with its home agent) is small, few datagrams will be lost. Recall that we saw in Chapter 3 that end-to-end connections can suffer datagram loss due to network congestion. Hence the occasional datagram loss within a connection when a node moves between networks is by no means a catastrophic problem. If loss-free communication is required, upper-layer mechanisms will recover from datagram loss, whether such loss results from network congestion or from user mobility.

An indirect routing approach is used in the mobile IP standard [RFC 3220], as discussed in Section 4.9.3.

Direct Routing to a Mobile Node

The indirect routing approach illustrated in Figure 4.60 suffers from an inefficiency known as the **triangle routing problem**—datagrams addressed to the mobile node must be routed first to the home agent and then to the foreign network, even when a much more efficient route exists between the correspondent and the mobile node. In the worst case, imagine a mobile user who is visiting the foreign network of a colleague. The two are sitting side-by-side and exchanging data over the network. Datagrams from the correspondent (in this case the colleague of the visitor) are routed to the mobile user's home agent and then back again to the foreign network!

Direct routing overcomes the inefficiency of triangle routing, but does so at the cost of additional complexity. In the direct routing approach, a **correspondent agent** in the correspondent's network first learns the COA of the mobile node. (It is also possible for the correspondent itself to perform the function of the correspondent agent, just as a mobile node could perform the function of the foreign agent.) This is shown as Steps 1 and 2 in Figure 4.62. The correspondent agent then tunnels datagrams directly to the mobile node's COA, in a manner analogous to the tunneling performed by the home agent, Steps 3 and 4 in Figure 4.62.

While direct routing overcomes the triangle routing problem, it introduces additional complexity. A protocol is needed for the correspondent agent to learn the mobile node's COA (Steps 1 and 2 in Figure 4.62). Another complication arises when the mobile node moves from one foreign network to another. One solution here is to create a new protocol to notify the correspondent of the changing COA. An alternative solution is for the new foreign agent to provide the old foreign agent with the mobile node's new COA. A foreign agent receiving an encapsulated datagram for a departed mobile node will reencapsulate the datagram and forward it using the new COA. If the mobile node moves through multiple networks, a chain of

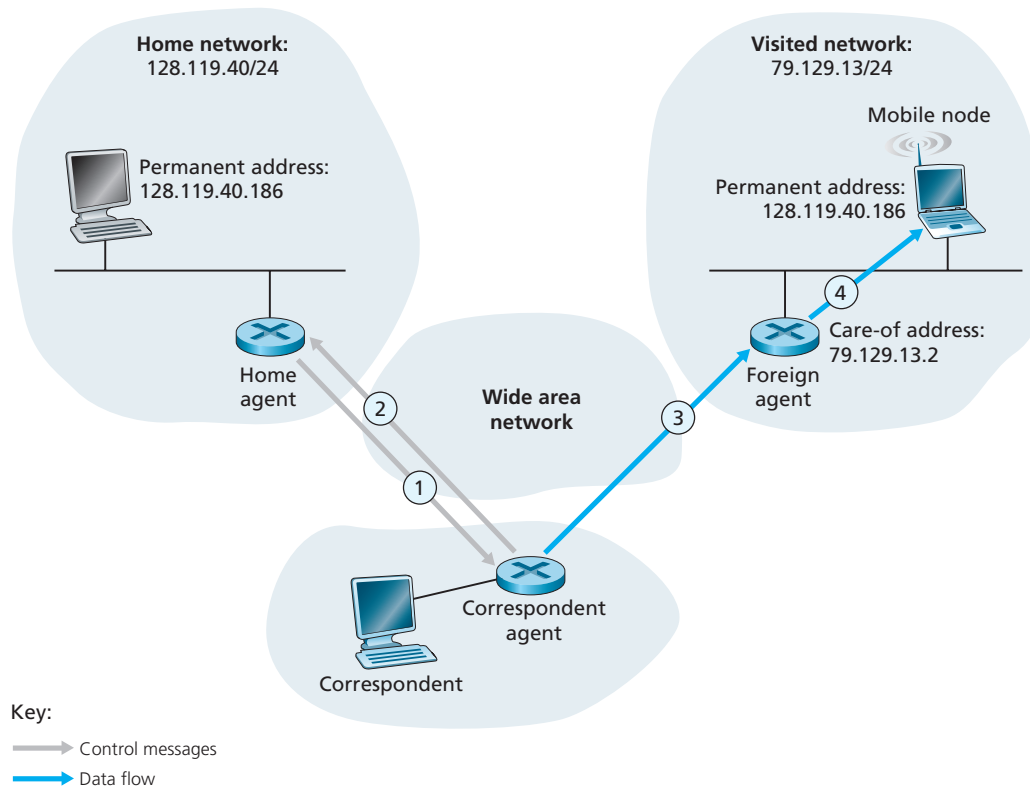


Figure 4.62 ♦ Direct routing to a mobile user

forwarding foreign agents will form. Thus, while this latter solution avoids the need to notify the correspondent about a changing COA, it requires significant additional coordination among the foreign agents.

A direct routing approach is taken in routing telephone calls to mobile users in several mobile telephone network standards, including GSM [Lin 2001]. An extension of the mobile IP standard to include direct routing is also under consideration [Perkins 2002].

4.9.3 Mobile IP

The Internet architecture and protocols for supporting mobility, collectively known as mobile IP, are defined primarily in RFC 3220. Mobile IP is a flexible standard, supporting many different modes of operation, for example, operation with or without a foreign agent, multiple ways for agents and mobile nodes to discover each other, use of single or multiple COAs, and multiple forms of encapsulation. As such, mobile IP is a complex standard, and would require an entire book to describe in

detail. Our modest goal here is to provide an overview of the most important aspects of mobile IP and to illustrate its use in a few common-case scenarios.

The mobile IP architecture contains many of the elements we have considered above, including the concepts of home agents, foreign agents, care-of addresses, and encapsulation/decapsulation. The current standard [RFC 3220] specifies the use of indirect routing to the mobile node, although draft modifications have been studied.

The mobile standard consists of three main pieces:

- ◆ **Agent discovery.** Mobile IP defines protocols used by a home or foreign agent to advertise its services to mobile nodes, and protocols for mobile nodes to solicit the services of a foreign or home agent.
- ◆ **Registration with the home agent.** Mobile IP defines the protocols used by the mobile node and/or foreign agent to register and deregister COAs with a mobile node's home agent.
- ◆ **Indirect routing of datagrams.** The standard also defines the manner in which datagrams are forwarded to mobile nodes by a home agent, including rules for forwarding datagrams, rules for handling error conditions, and several different forms of encapsulation [RFC 2003, RFC 2004].

Security considerations are prominent throughout the mobile IP standard. For example, authentication of a mobile node is clearly needed to ensure that a malicious user does not register a bogus care-of address with a home agent, which could cause all datagrams addressed to an IP address to be redirected to the malicious user. Mobile IP achieves security using many of the mechanisms that we will examine in Chapter 7, and so we will not address security considerations in our discussion below.

Agent Discovery

A mobile IP node arriving to a new network, whether attaching to a foreign network or returning to its home network, must learn the identity of the corresponding foreign or home agent. Indeed it is the discovery of a new foreign agent, with a new network address, that allows the network layer in a mobile node to learn that it has moved into a new foreign network. This process is known as **agent discovery**. Agent discovery can be accomplished in one of two ways: via agent advertisement or via agent solicitation.

With **agent advertisement**, a foreign or home agent advertises its services using an extension to the existing router discovery protocol [RFC 1256]. The agent periodically broadcasts an ICMP message with a type field of 9 (router discovery) on all links to which it is connected. The router discovery message contains the IP address of the router (that is, the agent), thus allowing a mobile node to learn the agent's IP address. The router discovery message also contains a mobility agent advertisement extension that contains additional information needed by the mobile node. Among the more important fields in the extension are:

- ◆ *Home agent bit (H)*: indicates that the agent is a home agent for the network in which it resides.
- ◆ *Foreign agent bit (F)*: indicates that the agent is a foreign agent for the network in which it resides.
- ◆ *Registration required bit (R)*: indicates that a mobile user in this network *must* register with a foreign agent. In particular, a mobile user cannot obtain a care-of address in the foreign network (for example, using DHCP) and assume the functionality of the foreign agent for itself, without registering with the foreign agent.
- ◆ *M, G encapsulation bits*: indicate whether a form of encapsulation other than IP-in-IP encapsulation will be used.
- ◆ *Care-of addresses (COA) fields*. A list of one or more care-of addresses provided by the foreign agent. In our example below, the COA will be associated with the foreign agent, who will receive datagrams sent to the COA and then forward them to the appropriate mobile node. The mobile user will select one of these addresses as its COA when registering with its home agent.

Figure 4.63 illustrates some of the key fields in the agent advertisement message.

With **agent solicitation**, a mobile node wanting to learn about agents without waiting to receive an agent advertisement can broadcast an agent solicitation message, which is simply an ICMP message with type value 10. An agent receiving the

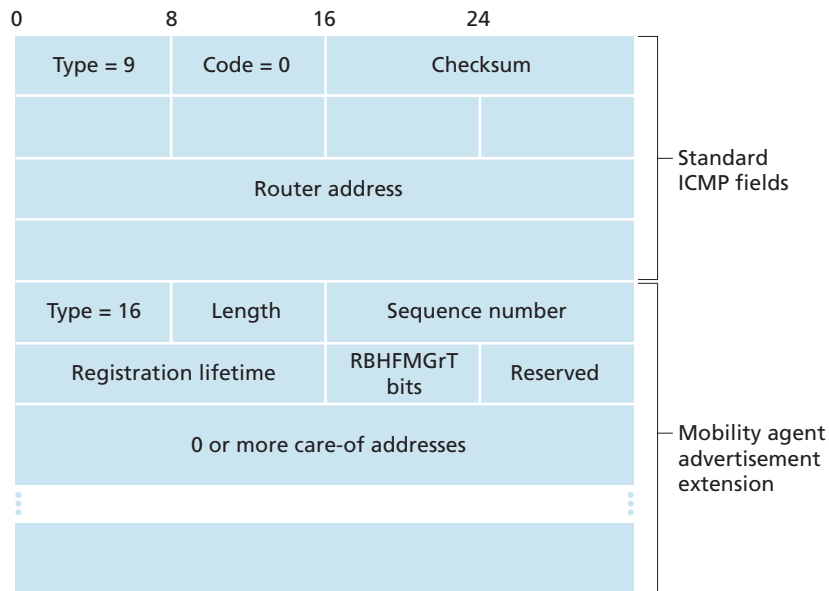


Figure 4.63 ◆ ICMP router discovery message with mobility agent advertisement extension

solicitation will unicast an agent advertisement directly to the mobile node, which can then proceed as if it had received an unsolicited advertisement.

Registration with the Home Agent

Once a mobile IP node has received a COA, that address must be registered with the home agent. This can be done either via the foreign agent (who then registers the COA with the home agent) or directly by the mobile IP node itself. We consider the former case below. Four steps are involved.

- ◆ Following the receipt of a foreign agent advertisement, a mobile node sends a mobile IP registration message to the foreign agent. The registration message is carried within a UDP datagram and sent to port 434. The registration message carries a COA advertised by the foreign agent, the address of the home agent (HA), the permanent address of the mobile node (MA), the requested lifetime of the registration, and a 64-bit registration identification. The requested registration lifetime is the number of seconds that the registration is to be valid. If the registration is not renewed at the home agent within the specified lifetime, the registration will become invalid. The registration identifier acts like a sequence number and serves to match a received reservation reply with a reservation request, as discussed below.
- ◆ The foreign agent receives the registration message and records the mobile node's permanent IP address. The foreign agent now knows that it should be looking for datagrams containing an encapsulated datagram whose destination address matches the permanent address of the mobile node. The foreign agent then sends a mobile IP registration message (again, within a UDP datagram) to port 434 of the home agent. The message contains the COA, HA, MA, encapsulation format requested, requested registration lifetime, and registration identification.
- ◆ The home agent receives the registration request and checks for authenticity and correctness. The home agent binds the mobile node's permanent IP address with the COA; in the future, datagrams arriving at the home agent and destined to the mobile node will now be encapsulated and tunneled to the COA. The home agents send a mobile IP registration reply containing the HA, MA, actual registration lifetime, and the registration identification of the request that is being satisfied with this reply.
- ◆ The foreign agent receives the registration reply and then forwards it to the mobile node.

At this point registration is complete, and the mobile node can receive datagrams sent to its permanent address. Figure 4.64 illustrates these steps. Note that the home agent specifies a lifetime that is smaller than the lifetime requested by the mobile node.

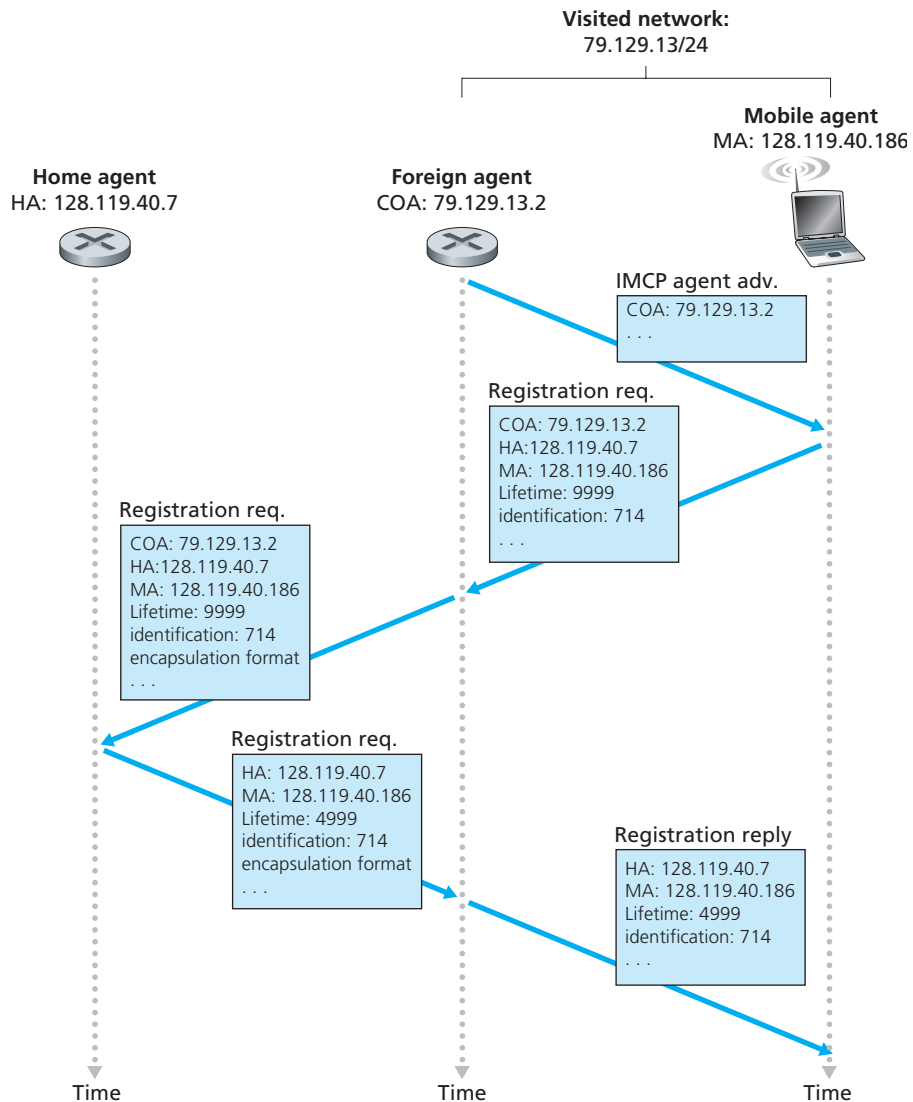


Figure 4.64 ♦ Agent advertisement and mobile IP registration

A foreign agent need not explicitly deregister a COA when a mobile node leaves its network. This will occur automatically, when the mobile node moves to a new network (whether another foreign network or its home network) and registers a new COA.

The mobile IP standard allows many additional scenarios and capabilities in addition to those described above. The interested reader should consult [Perkins 1998b; RFC 3220].

4.10 Summary

In this chapter, we began our journey into the network core. We learned that the network layer involves each and every host and router in the network. Because of this, network-layer protocols are among the most challenging in the protocol stack.

We learned that one of the biggest challenges in the network layer is routing datagrams through a network of millions of hosts and routers. We saw that this scaling problem is solved by partitioning large networks into independent administrative domains called autonomous systems (ASs). Each AS independently routes its datagrams through the AS, just as each country independently routes its postal mail through the country. In the Internet, two popular protocols for intra-AS routing are currently RIP and OSPF. To route packets among ASs, an inter-AS routing protocol is needed. The dominant inter-AS protocol today is BGP4.

Performing routing on two levels—one level for within each of the ASs and another level for among the ASs—is referred to as hierarchical routing. The scaling problem is largely solved by a hierarchical organization of the routing infrastructure. This is a general principle we should keep in mind when designing protocols, particularly for network-layer protocols: scaling problems can often be solved by hierarchical organization. It is interesting to note that this principle has been applied throughout the ages to many other disciplines besides computer networking, including corporate, government, religious, and military organizations.

In this chapter, we also learned about a second scaling issue: For large computer networks, a router may need to process millions of flows of packets between different source-destination pairs at the same time. To permit a router to process such a large number of flows, network designers have learned over the years that the router's tasks should be as simple as possible. Many measures can be taken to make the router's job easier, including using a datagram network layer rather than a virtual circuit network layer, using a streamlined and fixed-sized header (as in IPv6), eliminating fragmentation (also done in IPv6), and providing the one and only best-effort service. Perhaps the most important trick here is *not* to keep track of individual flows, but instead base routing decisions solely on hierarchical-structured destination addresses in the packets. It is interesting to note that the postal service has been using this same trick for many years.

In this chapter, we also looked at the underlying principles of routing algorithms. We learned that designers of routing algorithms abstract the computer network to a graph with nodes and links. With this abstraction, we can exploit the rich theory of shortest-path routing in graphs, which has been developed over the past 40 years in the operations research and algorithms communities. We saw that there are two broad approaches, a centralized approach, in which each node obtains a complete map of the network and independently applies a shortest-path routing algorithm; and a decentralized approach, in which individual nodes have only a partial picture of the entire network, yet the nodes work together to deliver packets along