1

Mobility on the Internet: Introduction

The best way to predict the future is to invent it. –Alan Kay

In order to understand what mobility on the Internet means, let us consider Figure 1.1. Bob, a user connected to the Internet by means of an access network (such as a cable modem or a DSL or a dial-up network), is "talking" to Alice, who is another user connected to the Internet by a different access network (e.g., WLAN). We do not show Bob and Alice themselves, but only their devices. Now, consider that Alice "moves" from her current access network to another access network (e.g., the Code Division Multiple Access (CDMA) cellular network). There are two basic problems we can see. First, how can Bob continue the existing conversation with Alice? Second, how can Bob reliably reach Alice once she has moved? The first problem can be broadly considered to be the *handover* problem. It is quite similar to that of users continuing their calls on their cell phones in spite of movement (such as in a moving train). The second is the *reachability* problem. It is quite similar to being able to reach users on their cell phones even when they are out of town (i.e., roaming). These two problems, which appear to be very straightforward, create many technical challenges associated with mobility on the Internet.

Perhaps it is tempting to assume that Mobile Internet is a given. Even so, it might be worthwhile to consider why the Internet Protocol is the best fit for a Mobile Internet. As we know, IP has been an unquestionable success in "gluing" disparate networks the world over. As the Internet becomes increasingly mobile, spearheaded by the

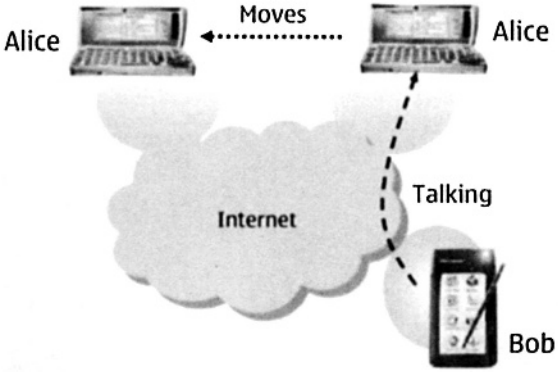

Fig. 1.1 Mobility on the Internet

mobile communication devices, users are bound to roam freely and attach to a variety of networks. In fact, this is already beginning to take place. Mobile Smartphones and Personal Digital Assistants equipped with cellular packet radio and WLAN are becoming all too common. In a Mobile Internet composed of numerous wireless (and wired) technologies, no single access network itself can provide mobility support across all kinds of networks. IP is best suited to address this imminent problem. More specifically, IP mobility is primarily a routing problem, but it includes problems involving fast switching of connections, smooth handovers, transport protocol optimizations, and many others.

The general problem of mobility is quite complicated, and no one protocol can be expected to provide a complete solution. In fact, just considering the problems introduced by mobility at the network layer (i.e., IP mobility), we still cannot expect to find a single protocol solution for all of them. These problems include the following:

1. Authentication for access rights to the new network
2. Obtaining an IP address at the new point of attachment
3. Routing packets to each new point of attachment
4. During transitions between points of attachment, taking appropriate actions to minimize handover delay and data loss not directly attributable to unavailability of the underlying communications medium
5. When packet is to be transmitted, selecting the appropriate source IP address

Solving the first two of these problems without attempting to solve the other problems can be said to provide a solution for portable computing. In other words, one can establish a link and start to work at a new point of attachment, but the user experience more closely resembles rebooting than relocating. Whatever activities were in progress before the new link is established may be aborted or start behaving unpredictably. In this book, making use of this kind of portable computing will be called *roaming*.

Solving the third problem is often related to network-layer protocol considerations because the act of supplying a new IP address depends on being able to prove that one is authorized to receive it. There are many different preexisting solutions to the problem of access control and authorizations. Many or even most of them are challenging to adapt to the needs of network-layer mobility. To authorize roaming is not so demanding, because, by the above definitions, the mobile user is already expecting significant disruption of whatever activities (if any) might have been in progress on the mobile device. For roaming scenarios, then, we can expect the access control to proceed without any dependency on mobility management, if indeed there is any mobility management. Once the access control is granted, the IP address can be allocated and the routing (or bridging, even) enabled for the particular device.

Roaming in this way is particularly easy for devices that are not named within the global *Domain Name System* or DNS [1, 2]. This is because there is no need to update the DNS entry for such devices, even when they receive new IP addresses [1]. On the other hand, for devices that do have a persistent name that is published by DNS, getting a new IP address immediately invalidates the DNS entry and effectively makes the device unreachable for incoming transmissions that are based on resolving the device's persistent name. In order to solve this problem, methods have been specified (see, for instance, [3]).

It is worthwhile to emphasize that updating DNS carries with it an obligation to provide strong proof that the update is properly authorized. Otherwise, if DNS were to accept such update requests without sufficient assurance about the identity of the requesting node, havoc would ensue. In fact, in today's Internet, these secure update procedures are not so widely deployed, and effectively must be considered to be unavailable to the majority of roaming nodes.

In this way, we can see already that the restrictions imposed by focusing only on the roaming problem have the effect of relegating roaming nodes to the status of "second-class citizens" of the Internet. They can operate as clients of well-known services (typically including e-mail services and browsing web pages that are statically addressable). But, for instance, such devices cannot by themselves receive telephone calls by way of the Internet, or publish web pages, or operate in true peer-to-peer fashion. For instance, Bob (in Figure 1.1) can neither reach Alice reliably nor maintain his call once she moves to a different network. Many people are content with these restrictions, because right now web browsing is the dominant application, and e-mail is not closely tied to real-time interactions.

[1] A DNS entry is required in order for a node to determine the IP address of its peer with which it wishes to communicate. The DNS contains the IP addresses for entries such as www.yahoo.com or any user record

We can distinguish between roaming and handover by observing that handover usually means arranging for transfer of control or responsibility. In the case of handover for a mobile network device, this means that a new network entity has to begin taking over some of the functions currently operating at some other network entity that is interacting with the mobile device. When the handover completes, all interactions are handled by the new entity, and the previous one can deallocate whatever resources were assigned for use with the mobile device.

That's all very abstract. If we think of network entities as access routers, then they are supposed to be delivering packets to the mobile device and forwarding traffic from the mobile device to its peers. From the standpoint of IP networking, the basic responsibility of the access routers is to forward packets to their destination, as indicated by the IP address of the destination. However, the access routers only have this responsibility for the addresses that are *topologically-correct* for their place in the network, and when the access routers are in different places, their customers will be expected to have different IP addresses. From this perspective, the access routers do not really offer any handover features to the mobile device unless they have been augmented with extra functions, such as those described in later chapters.

Using the same idea, we can think about designing other sorts of mobility-related handovers. For instance, one could handover a session from one display device to another. A new display controller would have to take over responsibility for showing the pixels and a new path created for transmitting the picture information to the display controller. If the devices are on different hardware platforms, some translation has to occur so that the picture information is presented to the display controller in a way that is suitable for the hardware. Perhaps the size of the screen is different, or perhaps one screen has fewer colors than the other. If the displays are attached to different points of the network, then some communications protocol has to be defined as a way of transferring the picture information to the new network node. If this is done carefully, one could well imagine a smooth handover with no frames lost. Perhaps with a lot of buffering, there could be a time when the same images are streaming to both display controllers, so that the viewer would not miss a single dot.

Perhaps it is useful to discuss topological correctness of IP addresses a little bit. The Internet addressing is often termed as *hierarchical* meaning that organizations typically get chunks of IP addresses which are valid for all the hosts attached to *that* particular segment of the Internet. In order to maintain its (amazing) scalability, the Internet routing works based on the network prefixes and not on individual host IP addresses: only when a packet reaches the destination subnet does the host-based forwarding take place. This means when a host with an IP address IP_A valid on network A moves to network B with its own set of addresses on the Internet topology, the Internet routing cannot ensure forwarding of packets to the host's IP address IP_A. The host must obtain another IP address on network B, and then everything moves like clockwork. In other words, a host address must be *topologically-consistent* or *topologically-correct* for packets to reach it.

In order to tackle the problem of change of IP address, Mobile IP provides a way for a mobile device to maintain a persistent IP address at the same time that it acquires a new IP address from every new access network that it visits. The mobile device may be viewed as taking some of the responsibility for overcoming the effects that result from changing its IP address by shielding the change from its applications. Since the applications don't see any change of address, they continue without major disruption when the mobile device acquires a new IP address at its new point of attachment.

Mobile IP allows the mobile device to remain addressable at its persistent IP address, known as the device's *Home Address*. This is the address used by the applications running on the mobile device, and it is the address registered in the global DNS. Since the address doesn't change, the need for secure updates to DNS is sidestepped completely. That is already a major benefit. By the same token, existing Transmission Control Protocol (TCP) connections can survive relocation to a new point of attachment, since Mobile IP allows the device to continue using its persistent address even when the device is only reachable at an address other than its persistent address. Such a "roaming address" is called as *Care-of Address* in Mobile IP. Both of these addresses are known to the mobile node as well as its trusted partner called the *Home Agent*, which can be considered as a router on the mobile node's *home network*, the IP subnet that the mobile node normally resides in when it is not roaming. In a way, quite a bit of the Mobile IP protocol is between the mobile node and its home agent. The rest of the protocol is between the mobile node and its *Correspondent Node*, which is any Internet-enabled device that communicates with the mobile node.

A good portion of this book describes the essentials of the Mobile IP protocol in the context of IPv6. The other parts are dedicated to protocols that provide the necessary performance for real-time applications such as Voice over IP (VoIP). In order to fully understand these protocols, it is necessary to understand the basics of IPv6 and some elements of the IP security (IPsec) architecture. In the next two chapters, we provide a brief introduction to these two topics, expecting the users to consult references elsewhere for a much more detailed discussion. We will return to the detailed description of the Mobile IP protocol in Part II.

The *Request For Comments* (RFCs) in References in this chapter and the rest of the book can be accessed via http://ietf.org/rfcs/html.

REFERENCES

1. P. Mockapetris, "Domain Names - Concepts and Facilities," RFC 1034, Internet Engineering Task Force, November 1987.

2. P. Mockapetris, "Domain Names - Implementation and Specification," RFC 1035, Internet Engineering Task Force, November 1987.

3. B. Wellington, "Secure Domain Name System Dynamic Update," RFC 3007, Internet Engineering Task Force, November 2000.