20.1 Introduction

In this chapter, we will describe *broadcasting* and in the next chapter, we will describe *multicasting*. All the examples in the text so far have dealt with *unicasting:* a process talking to exactly one other process. Indeed, TCP works with only unicast addresses, although UDP and raw IP support other paradigms. Figure 20.1 shows a comparison of the different types of addressing.

Figure 20.1. Different forms of addressing.

IPv6 has added *anycasting* to the addressing architecture. An IPv4 version of anycasting, which was never widely deployed, is described in RFC 1546 [Partridge, Mendez, and Milliken 1993]. IPv6 anycasting is defined in RFC 3513 [Hinden and Deering 2003]. Anycasting allows addressing one (usually the "closest" by some metric) system out of a set of systems that usually provides identical services. With an appropriate routing configuration, hosts can provide anycasting services in either IPv4 or IPv6 by injecting the same address into the routing protocol in multiple locations. However, RFC 3513's anycasting only permits routers to have anycast addresses; hosts may not provide anycasting services. As of this writing, there is no API defined for using anycast addresses. There is work in progress to refine the IPv6 anycast architecture, and hosts may be able to dynamically provide anycasting services in the future.

The important points in Figure 20.1 are:

- Multicasting support is optional in IPv4, but mandatory in IPv6.
- Broadcasting support is not provided in IPv6. Any IPv4 application that uses broadcasting must be recoded for IPv6 to use multicasting instead.
- Broadcasting and multicasting require datagram transport such as UDP or raw IP; they cannot work with TCP.

One use for broadcasting is to locate a server on the local subnet when the server is assumed to be on the local subnet but its unicast IP address is not known. This is sometimes called *resource discovery*. Another use is to minimize the network traffic on a LAN when there are multiple clients communicating with a single server. There are numerous examples of Internet applications that use broadcasting for this purpose. Some of these can also use multicasting.

- ARP— Although this is a protocol that lies underneath IPv4, and not a user application, ARP broadcasts a request on the local subnet that says, "Will the system with an IP address of a.b.c.d please identify yourself and tell me your hardware address?" ARP uses link-layer broadcast, not IP-layer, but is an example of a use of broadcasting.
- DHCP— The client assumes a server or relay is on the local subnet and sends its request to the broadcast address (often 255.255.255.255 since the client doesn't yet know its IP address, its subnet mask, or the limited broadcast address of the subnet).
- Network Time Protocol (NTP)— In one common scenario, an NTP client is configured with the IP address of one or more servers to use, and the client polls the servers at some frequency (every 64 seconds or longer). The client updates its clock using sophisticated algorithms based on the time-of-day returned by the servers and the RTT to the servers. But on a broadcast LAN, instead of making each of the clients poll a single server, the server can broadcast the current time every 64 seconds for all the clients on the local subnet, reducing the amount of network traffic.
- Routing daemons— The oldest routing daemon, `routed`, which implements RIP version 1, broadcasts its routing table on a LAN. This allows all other routers attached to the LAN to receive these routing announcements, without each router having to be configured with the IP addresses of all its neighboring routers. This feature can also be used by hosts on the LAN listening to these routing announcements and updating their routing tables accordingly. RIP version 2 permits the use of either multicast or broadcast.

We must note that multicasting can replace both uses of broadcasting (resource discovery and reducing network traffic) and we will describe the problems with broadcasting later in this chapter and the next chapter.