Chapter 12. For example, if the routers in the intranet of Figure 14.2 on page 291 use RIP, router R_1 will broadcast a message on network *2* that contains the pair (*1*, *1*), meaning that it can reach network *1* at distance (i.e., cost) *1*. Routers R_2 and R_5 will receive the broadcast and install a route to network *1* through R_1 (at cost *2*). Later, routers R_2 and R_5 will include the pair (*1*, *2*) when they broadcast their RIP messages on network *3*. Eventually, all routers will install a route to network *1*.

RIP specifies a few rules to improve performance and reliability. For example, once a router learns a route from another router, it must apply *hysteresis*, meaning that it does not replace the route with an equal cost route. In our example, if routers R_2 and R_5 both advertise network *1* at cost *2*, routers R_3 and R_4 will install a route through the one that happens to advertise first. We can summarize:

> *To prevent oscillation among equal cost paths, RIP specifies that existing routes should be retained until a new route has strictly lower cost.*

What happens if a router fails (e.g., the router crashes)? RIP specifies that when a router receives and installs a route in its forwarding table, the router must start a timer for the entry. The timer is reset whenever the router receives another RIP message advertising the same route. The route becomes invalid if 180 seconds pass without the route being advertised again.

RIP must handle three kinds of errors caused by the underlying algorithm. First, because the algorithm does not explicitly detect forwarding loops, RIP must either assume participants can be trusted or take precautions to prevent such loops. Second, to prevent instabilities, RIP must use a low value for the maximum possible distance (RIP uses *16*). Thus, for intranets in which legitimate hop counts approach *16*, managers must divide the intranet into sections or use an alternative protocol†. Third, the distance-vector algorithm used by RIP can create a problem known as *slow convergence* or *count to infinity* in which inconsistencies arise because routing update messages propagate slowly across the network. Choosing a small infinity (*16*) helps limit slow convergence, but does not eliminate it.

14.4 Slow Convergence Problem

Forwarding table inconsistencies and the slow convergence problem are not unique to RIP. They are fundamental problems that can occur with any distance-vector protocol in which update messages carry only pairs of destination network and distance to that network. To understand the problem, consider using a distance-vector protocol on the routers in Figure 14.2 (page 291). To simplify the example, we will only consider three routers, R_1, R_2, and R_3, and only consider the routes they have for network *1*. To reach network *1*, R_3 forwards to R_2, and R_2 forwards to R_1. Part (a) of Figure 14.4 illustrates the forwarding.

---

†Note that the hop count used in RIP measures the *span* of the intranet — the longest distance between two routers — rather than the total number of networks or routers. Most corporate intranets have a span that is much smaller than 16.