NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

COMPUTER NETWORK AND ROUTING PROBLEM

Graduation Project
COM- 400

Student: MOHAMMED HUSEIN (20000499)

Supervisor: Prof. Dr Dogan Ibrahim

Nicosia - 2004

ACKNOWLEDGMENT

"First I would like to thank my God for giving me the privilege strength and power. I would like to give special thanks to my mother, she has supported me through out every step of my life for which I shall never forget it till the day I die and my father for without him I would have not been what I am today. On this note I would like to express my gratitude to Near East University for the knowledge that they supported me, and I would like to thank my teachers, I am forever in their gratitude for having the belief in me and not giving upon me. I am very grateful to all the people in my life, who have advised me, and who have always encouraged me to follow my dreams.

Secondly I would like to thank my supervisor Assoc. Prof. Dr Dogan Ibrahim for supervising my work. Under the guidance of him. I successfully overcome many difficulties and learned a lot about Computer Network and Routing Problem. In each discussion, he used to explain and answer. While teaching, he always helped me a lot either in my study or my life, and I felt my quick progress from his advises.

Special thanks for my friend Abdulmajeed Jaffer for supporting me and to complete my project and my roommate Mahmoud Abuhmaid and all my friends in Nezer Apartment.

Finally, I would like to thank all the people who support me all the time and for spending a nice time with them in this university and in this country.

ABSTRACT

Routing is the act of moving information across an inter network from a source to a destination. Along the way, at least one intermediate node typically is encountered. Routing is often contrasted with bridging, which might seem to accomplish precisely the same thing to the casual observer.

Routing involves two basic activities: determining optimal routing paths and transporting information groups (typically called packets) through an inter network. In the context of the routing process, the latter of these is referred to as switching. Although switching is relatively straightforward, path determination can be very complex.

Routers are devices that implement the network service. They provide interfaces for a wide range of links and sub networks at a wide range of speeds. Routers are active and intelligent network nodes that can participate in managing a network.

Routers are required to support multiple protocol stacks, each with its own routing protocols, and to allow these different environments to operate in parallel. In practice, routers also incorporate bridging functions and sometimes serve as a limited form of hub.

INTRODUCTION

In the first chapter we talk about the basics of routers and what are the components of routing, the path determination, how switching works and how Routing Algorithms work. We also see what type of algorithms are used.

Here in the second chapter we talk on why protocols in routing are necessary, what are static and dynamic routing and what are the uses of these two types of routing. We also cover on how distances on network paths are determined by various metrics. The time of convergence is also covered.

Chapter three moves us to the distance vector routing basics, here we see on how distance vector protocols exchange routing tables work, How Topology changes Propagate Through The NetWork Of Routers and what are the problems of routing loops along with the problem of counting to infinity and the solutions of defining a maximum, split horizon and hold down timers.

In chapter four covers link state routing, it covers on what are the key characteristics, what are the processing and memory requirements along with the bandwitdh requirements. The change of topologies though the network of routers is also covered.

Chapter five takes us to the different types of routing protocols used such as Distance-Vector protocol, Link-State protocol and Hybrid routing protocols. We also talk about the differnce between the distance vector and link state routing protocol. How a router selects its path and how a router switches to multiple protocols is also explained. This chapter also explain on how LAN to LAN and LAN to WAN routing works.

Chapter six talk on the initial router configuration, how to check the router for any problems and how to solve it. The intial routing table is also covered. It also covers on how a router learns about its destinations. The IP route command is also explained.

In chapter seven Interior Gateway Routing Protocol (IGRP) is explained along with the IGRP protocol characteristics and what are the stability features. We also see on how an IGRP protocol uses an updated, invalid and hold-time timer.

Chapter eight takes us to the topic of Open Shortest Path First (OSPF) and what are the routing hierarchy it uses and the type of algorithim used. The shortest path first (SPF) algorithim along with the packet form and the additonal feautures of the OSPF is explained.

Chapter nine covers Enhanced Interior Gateway Routing Protocol (EIGRP), what are the Capabilities and Attributes, the type of tables is uses. The underlying process and technologies with the neighbouring tables are also explained, the topologies of tables, routing states

Chapter ten which is the last chapter covers the Border Gateway Protocol (BGP). The operations, routing, message types, packet formats, Packet-Header Fields, Open Message Fields and the Notification Message Fields.

CHAPTER ONE

Routing Basics

1.1 What Is Routing?

Routing is the act of moving information across an inter network from a source to a destination. Along the way, at least one intermediate node typically is encountered. Routing is often contrasted with bridging, which might seem to accomplish precisely the same thing to the casual observer. The primary difference between the two is that bridging occurs at Layer 2 (the link layer) of the OSI reference model, whereas routing occurs at Layer 3 (the network layer). This distinction provides routing and bridging with different information to use in the process of moving information from source to destination, so the two functions accomplish their tasks in different ways.

The topic of routing has been covered in computer science literature for more than two decades, but routing achieved commercial popularity as late as the mid-1980s. The primary reason for this time lag is that networks in the 1970s were fairly simple, homogeneous environments. Only relatively recently has large-scale inter networking become popular.

1.2 Routing Components

Routing involves two basic activities: determining optimal routing paths and transporting information groups (typically called packets) through an inter network. In the context of the routing process, the latter of these is referred to as switching. Although switching is relatively straightforward, path determination can be very complex.

1.3 Path Determination

Path determination, for traffic going through a network cloud, occurs at the network layer (Layer 3). The path determination function enables a router to evaluate the available paths to a destination and to establish the preferred handling of a packet. Routing services use network topology information when evaluating network paths. This information can be configured by the network administrator or collected through dynamic processes running in the network.

The network layer provides best-effort end-to-end packet delivery across interconnected networks. The network layer uses the IP routing table to send packets from the source network to the destination network. After the router determines which path to use, it proceeds with forwarding the packet. It takes the packet that it accepted on one interface and forwards it to another interface or port that reflects the best path to the packet's destination.

A metric is a standard of measurement, such as path length, that is used by routing algorithms to determine the optimal path to a destination. To aid the process of path determination, routing algorithms initialize and maintain routing tables, which contain route information. Route information varies depending on the routing algorithm used.

Routing algorithms fill routing tables with a variety of information. Destination/next hop associations tell a router that a particular destination can be gained optimally by sending the packet to a particular router representing the "next hop" on the way to the final destination. When a router receives an incoming packet, it checks the destination address and attempts to associate this address with a next hop. Figure 1-1 depicts a sample destination/next hop routing table.

To reach network:	Send to:
27	Node A
57	Node B
17	Node C
24	Node A
52	Node A
16	Node B
26	Node A
.	.
.	.
.	.

Figure 1-1: Destination/next hop associations determine the data's optimal path.

Routing tables also can contain other information, such as data about the desirability of a path. Routers compare metrics to determine optimal routes, and these metrics differ depending on the design of the routing algorithm used. A variety of common metrics will be introduced and described later in this chapter.

Routers communicate with one another and maintain their routing tables through the transmission of a variety of messages as shown in figure 1-2. The routing update message is one such message that generally consists of all or a portion of a routing table. By analyzing routing updates from all other routers, a router can build a detailed picture of network topology. A link-state advertisement, another example of a message sent between routers, informs other routers of the state of the sender's links. Link information also can be used to build a complete picture of topology to enable routers to determine optimal routes to network destinations.

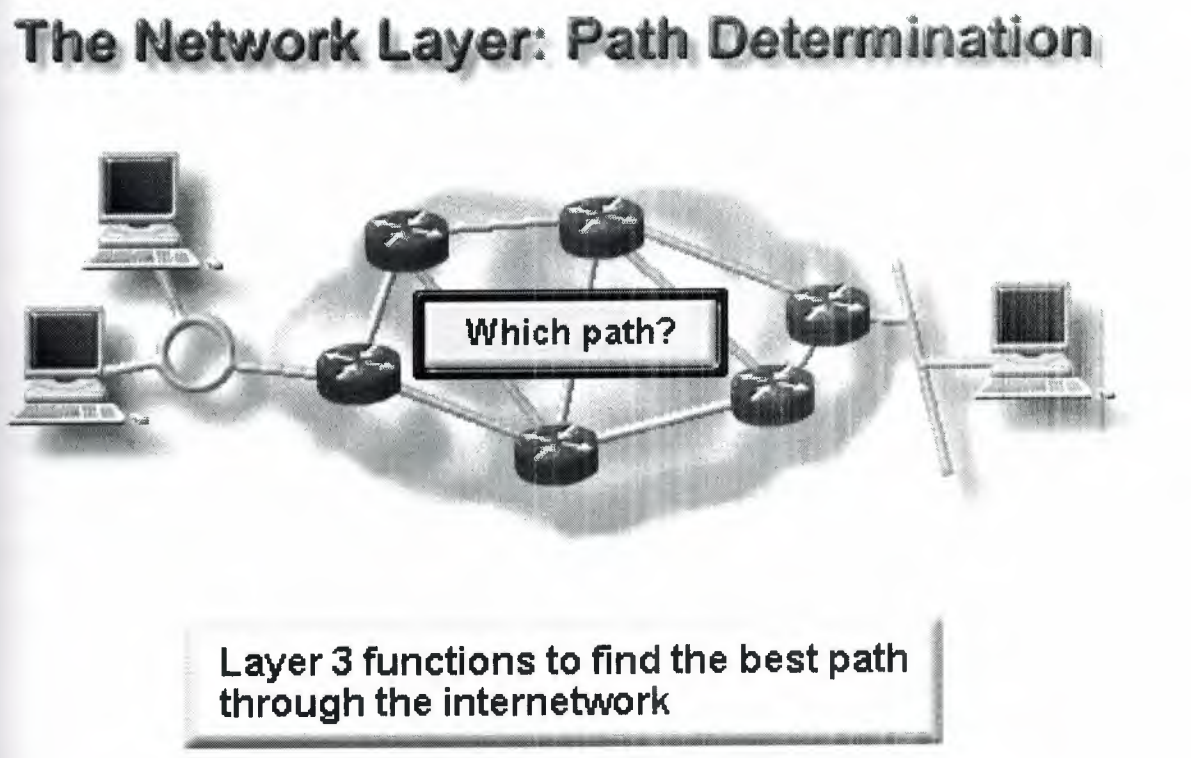

Figure 1-2: Path determination

1.4 How outers Route Packets From Source To Destination

To be truly practical, a network must consistently represent the paths available between routers. As Figure 1-2 shows, each line between the routers has a number that the routers use as a network address. These addresses must convey information that can be used by a routing process to pass packets from a source toward a destination. Using these addresses, the network layer can provide a relay connection that interconnects independent networks.

The consistency of Layer 3 addresses across the entire inter network also improves the use of bandwidth by preventing unnecessary broadcasts. Broadcasts invoke unnecessary process overhead and waste capacity on any devices or links that do not need to receive the broadcasts.

By using consistent end-to-end addressing to represent the path of media connections, the network layer can find a path to the destination without unnecessarily burdening the devices or links on the inter network with broadcasts.

1.5 Switching

Switching algorithms are relatively simple and are basically the same for most routing protocols. In most cases, a host determines that it must send a packet to another host. Having acquired a router's address by some means, the source host sends a packet addressed specifically to a router's physical (Media Access Control [MAC]-layer) address, this time with the protocol (network- layer) address of the destination host.

As it examines the packet's destination protocol address, the router determines that it either knows or does not know how to forward the packet to the next hop. If the router does not know how to forward the packet, it typically drops the packet. If the router knows how to forward the packet, it changes the destination physical address to that of the next hop and transmits the packet.

The next hop may, in fact, be the ultimate destination host. If not, the next hop is usually another router, which executes the same switching decision process. As the packet moves through the internetwork, its physical address changes, but its protocol address remains constant as shown in figure 1-3.

The preceding discussion describes switching between a source and a destination end system. The International Organization for Standardization (ISO) has developed a hierarchical terminology that is useful in describing this process. Using this terminology, network devices without the capability to forward packets between subnetworks are called end systems (ESs), whereas network devices with these capabilities are called intermediate systems (ISs).

ISs are further divided into those that can communicate within routing domains (intradomain ISs) and those that communicate both within and between routing domains (interdomain ISs). A routing domain generally is considered to be a portion of an internetwork under common administrative authority that is regulated by a particular set of administrative guidelines. Routing domains are also called autonomous systems. With certain protocols, routing domains can be divided into routing areas, but intradomain routing protocols are still used for switching both within and between areas.

A router generally relays a packet from one data link to another, using two basic functions:

- a path determination function
- a switching function.

The router uses addressing for these routing and switching functions. The router uses the network portion of the address to make path selections to pass the packet to the next router along the path.

The switching function allows a router to accept a packet on one interface and forward it through a second interface. The path determination function enables the router to select the most appropriate interface for forwarding a packet. The node portion of the address is used by the final router (the router connected to the destination network) to deliver the packet to the correct host.

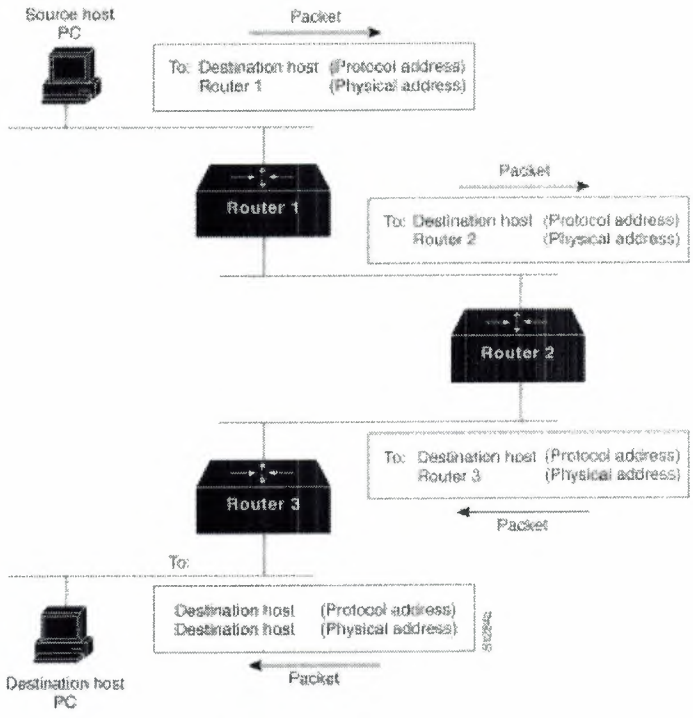

Figure 1.3: Numerous routers may come into play during the switching process.

1.6 Routed Versus Routing Protocol

Because of the similarity of the two terms, confusion often exists with routed protocol and routing protocol.

Routed protocol is any network protocol that provides enough information in its network layer address to allow a packet to be forwarded from one host to another host based on the addressing scheme. Routed protocols define the field formats within a packet. Packets are generally conveyed from end system to end system. The Internet Protocol (IP) is an example of a routed protocol.

Routing protocols support a routed protocol by providing mechanisms for sharing routing information. Routing protocol messages move between the routers.

A routing protocol allows the routers to communicate with other routers to update and maintain tables. TCP/IP examples of routing protocols are:

- RIP (Routing Information Protocol)
- IGRP (Interior Gateway Routing Protocol)
- EIGRP (Enhanced Interior Gateway Routing Protocol)

1.7 Multiprotocol Routing

Routers are capable of supporting multiple independent routing protocols and maintaining routing tables for several routed protocols. This capability allows a router to deliver packets from several routed protocols over the same data links as shown in figure 1-4.

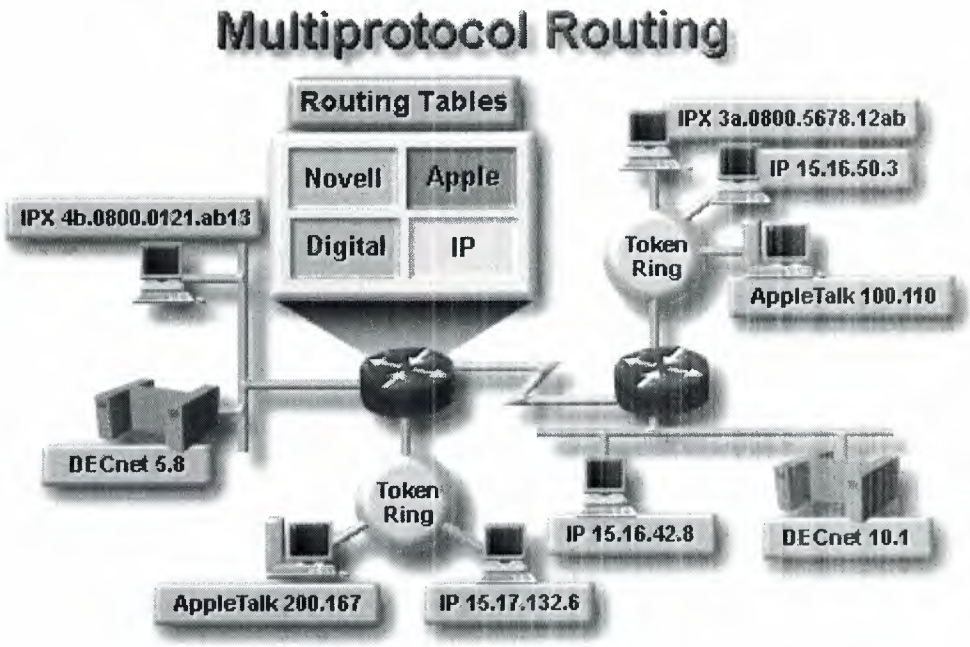

Figure1-4 Routers Path Traffic From All Routed Protocols Over The Internetwork

1.8 Routing Algorithms

Routing algorithms can be differentiated based on several key characteristics. First, the particular goals of the algorithm designer affect the operation of the resulting routing protocol. Second, various types of routing algorithms exist, and each algorithm has a different impact on network and router resources. Finally, routing algorithms use a variety of metrics that affect calculation of optimal routes. The following sections analyze these routing algorithm attributes.

1.9 Design Goals

Routing algorithms often have one or more of the following design goals:

Optimality
Simplicity and low overhead
Robustness and stability
Rapid convergence
Flexibility

Optimality refers to the capability of the routing algorithm to select the best route, which depends on the metrics and metric weightings used to make the calculation. One routing algorithm, for example, may use a number of hops and delays, but may weight delay more heavily in the calculation. Naturally, routing protocols must define their metric calculation algorithms strictly.

Routing algorithms also are designed to be as simple as possible. In other words, the routing algorithm must offer its functionality efficiently, with a minimum of software and utilization overhead. Efficiency is particularly important when the software implementing the routing algorithm must run on a computer with limited physical resources.

Routing algorithms must be robust, which means that they should perform correctly in the face of unusual or unforeseen circumstances, such as hardware failures, high load conditions, and

incorrect implementations. Because routers are located at network junction points, they can cause considerable problems when they fail. The best routing algorithms are often those that have withstood the test of time and have proven stable under a variety of network conditions.

In addition, routing algorithms must converge rapidly. Convergence is the process of agreement, by all routers, on optimal routes. When a network event causes routes either to go down or become available, routers distribute routing update messages that permeate networks, stimulating recalculation of optimal routes and eventually causing all routers to agree on these routes. Routing algorithms that converge slowly can cause routing loops or network outages.

In the routing loop displayed in Figure 1-5, a packet arrives at Router 1 at time t1. Router 1 already has been updated and thus knows that the optimal route to the destination calls for Router 2 to be the next stop. Router 1 therefore forwards the packet to Router 2, but because this router has not yet been updated, it believes that the optimal next hop is Router 1. Router 2 therefore forwards the packet back to Router 1, and the packet continues to bounce back and forth between the two routers until Router 2 receives its routing update or until the packet has been switched the maximum number of times allowed.

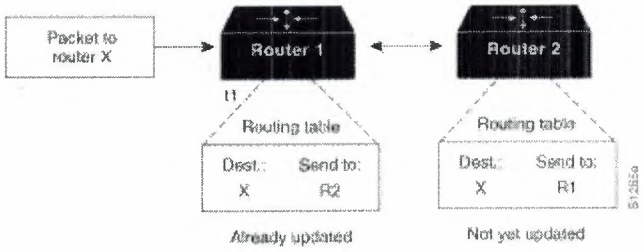

Figure 1-5: Slow convergence and routing loops can hinder progress.

Routing algorithms should also be flexible, which means that they should quickly and accurately adapt to a variety of network circumstances. Assume, for example, that a network segment has gone down. As they become aware of the problem, many routing algorithms will quickly select the next-best path for all routes normally using that segment. Routing algorithms can be programmed to adapt to changes in network bandwidth, router queue size, and network delay, among other variables.

1.10 Algorithm Types

Routing algorithms can be classified by type. Key differentiators include:

- Static versus dynamic
- Single-path versus multi-path
- Flat versus hierarchical
- Host-intelligent versus router-intelligent
- Intra domain versus inter domain
- Link state versus distance vector

1.10.1 Static Versus Dynamic

Static routing algorithms are hardly algorithms at all, but are table mappings established by the network administrator prior to the beginning of routing. These mappings do not change unless the network administrator alters them. Algorithms that use static routes are simple to design and work well in environments where network traffic is relatively predictable and where network design is relatively simple.

Because static routing systems cannot react to network changes, they generally are considered unsuitable for today's large, changing networks. Most of the dominant routing algorithms in the 1990s are dynamic routing algorithms, which adjust to changing network circumstances by analyzing incoming routing update messages. If the message indicates that a network change has occurred, the routing software recalculates routes and sends out new routing update messages. These messages permeate the network, stimulating routers to rerun their algorithms and change their routing tables accordingly.

Dynamic routing algorithms can be supplemented with static routes where appropriate. A router of last resort (a router to which all unroutable packets are sent), for example, can be

designated to act as a repository for all unroutable packets, ensuring that all messages are at least handled in some way.

1.10.2 Single-Path Versus Multipath

Some sophisticated routing protocols support multiple paths to the same destination. Unlike single-path algorithms, these multipath algorithms permit traffic multiplexing over multiple lines. The advantages of multipath algorithms are obvious: They can provide substantially better throughput and reliability.

1.10.3 Flat Versus Hierarchical

Some routing algorithms operate in a flat space, while others use routing hierarchies. In a flat routing system, the routers are peers of all others. In a hierarchical routing system, some routers form what amounts to a routing backbone. Packets from non-backbone routers travel to the backbone routers, where they are sent through the backbone until they reach the general area of the destination. At this point, they travel from the last backbone router through one or more non-backbone routers to the final destination.

Routing systems often designate logical groups of nodes, called domains, autonomous systems, or areas. In hierarchical systems, some routers in a domain can communicate with routers in other domains, while others can communicate only with routers within their domain. In very large networks, additional hierarchical levels may exist, with routers at the highest hierarchical level forming the routing backbone.

The primary advantage of hierarchical routing is that it mimics the organization of most companies and therefore supports their traffic patterns well. Most network communication occurs within small company groups (domains).

Because intradomain routers need to know only about other routers within their domain, their routing algorithms can be simplified, and, depending on the routing algorithm being used, routing update traffic can be reduced accordingly.

1.10.4 Host-Intelligent Versus Router-Intelligent

Some routing algorithms assume that the source end-node will determine the entire route. This is usually referred to as source routing. In source-routing systems, routers merely act as store-and-forward devices, mindlessly sending the packet to the next stop.

Other algorithms assume that hosts know nothing about routes. In these algorithms, routers determine the path through the inter network based on their own calculations. In the first system, the hosts have the routing intelligence. In the latter system, routers have the routing intelligence.

The trade-off between host-intelligent and router-intelligent routing is one of path optimality versus traffic overhead. Host-intelligent systems choose the better routes more often, because they typically discover all possible routes to the destination before the packet is actually sent. They then choose the best path based on that particular system's definition of "optimal." The act of determining all routes, however, often requires substantial discovery traffic and a significant amount of time.

1.10.5 Intra Domain Versus Inter Domain

Some routing algorithms work only within domains; others work within and between domains. The nature of these two algorithm types is different. It stands to reason, therefore, that an optimal intra domain- routing algorithm would not necessarily be an optimal inter domain-routing algorithm.

1.10.6 Link State Versus Distance Vector

Link- state algorithms (also known as shortest path first algorithms) flood routing information to all nodes in the inter network. Each router, however, sends only the portion of the routing table that describes the state of its own links. Distance- vector algorithms (also known as Bellman-Ford algorithms) call for each router to send all or some portion of its routing table, but only to its neighbors. In essence, link- state algorithms send small updates everywhere, while distance- vector algorithms send larger updates only to neighboring routers.

Because they converge more quickly, link- state algorithms are somewhat less prone to routing loops than distance- vector algorithms. On the other hand, link- state algorithms require more CPU power and memory than distance- vector algorithms. Link-state algorithms, therefore, can be more expensive to implement and support. Despite their differences, both algorithm types perform well in most circumstances.

1.11 Routing Metrics

Routing tables contain information used by switching software to select the best route. But how, specifically, are routing tables built? What is the specific nature of the information they contain? How do routing algorithms determine that one route is preferable to others?

Routing algorithms have used many different metrics to determine the best route. Sophisticated routing algorithms can base route selection on multiple metrics, combining them in a single (hybrid) metric. All the following metrics have been used:

- Path Length
- Reliability
- Delay
- Bandwidth
- Load
- Communication Cost

Path length is the most common routing metric. Some routing protocols allow network administrators to assign arbitrary costs to each network link. In this case, path length is the sum of the costs associated with each link traversed. Other routing protocols define *hop count*, a metric that specifies the number of passes through internetworking products, such as routers, that a packet must take en route from a source to a destination.

Reliability, in the context of routing algorithms, refers to the dependability (usually described in terms of the bit-error rate) of each network link. Some network links might go down more often than others. After a network fails, certain network links might be repaired more easily or more quickly than other links. Any reliability factors can be taken into account in the assignment of the reliability ratings, which are arbitrary numeric values usually assigned to network links by network administrators.

Routing delay refers to the length of time required to move a packet from source to destination through the inter network. Delay depends on many factors, including the bandwidth of intermediate network links, the port queues at each router along the way, network congestion on all intermediate network links, and the physical distance to be traveled. Because delay is a conglomeration of several important variables, it is a common and useful metric.

Bandwidth refers to the available traffic capacity of a link. All other things being equal, a 10-Mbps Ethernet link would be preferable to a 64-kbps leased line. Although bandwidth is a rating of the maximum attainable throughput on a link, routes through links with greater bandwidth do not necessarily provide better routes than routes through slower links. If, for example, a faster link is busier, the actual time required to send a packet to the destination could be greater.

Load refers to the degree to which a network resource, such as a router, is busy. Load can be calculated in a variety of ways, including CPU utilization and packets processed per second. Monitoring these parameters on a continual basis can be resource-intensive itself.

Communication cost is another important metric, especially because some companies may not care about performance as much as they care about operating expenditures. Even though line delay may be longer, they will send packets over their own lines rather than through the public lines that cost money for usage time.

1.12 Network Protocol

Routed protocols are transported by routing protocols across an inter network. In general, routed protocols in this context also are referred to as *network* protocols. These network protocols perform a variety of functions required for communication between user applications in source and destination devices, and these functions can differ widely among protocol suites. Network protocols occur at the upper four layers of the OSI reference model: the transport layer, the session layer, the presentation layer, and the application layer.

Confusion about the terms routed protocol and routing protocol is common. Routed protocols are protocols that are routed over an inter network. Examples of such protocols are the Internet Protocol (IP), DEC net, AppleTalk, Novell NetWare, OSI, Banyan VINES, and Xerox Network System (XNS). Routing protocols, on the other hand, are protocols that implement routing algorithms. Put simply, routing protocols direct protocols through an inter network. Examples of these protocols include Interior Gateway Routing Protocol (IGRP), Enhanced Interior Gateway Routing Protocol (Enhanced IGRP), Open Shortest Path First (OSPF), Exterior Gateway Protocol (EGP), Border Gateway Protocol (BGP), Intermediate System to Intermediate System (IS-IS), and Routing Information Protocol (RIP). Routed and routing protocols are discussed in detail later.

Routed protocol is any network protocol that provides enough information in its network layer address to allow a packet to be forwarded from one host to another host based on the addressing scheme user information.

Routed protocols define the field formats and use within a packet. Packets are generally conveyed from end system to end system. The Internet Protocol (IP) as shown in figure 1-5.

Routing protocol supports a routed protocol by providing mechanisms for sharing routing information. Routing protocol messages move between the routers. A routing protocol allows the routers to communicate with other routers to update and maintain tables. TCP/IP examples of routing protocols are:

- RIP (Routing Information Protocol)
- IGRP (Interior Gateway Routing Protocol)
- EIGRP (Enhanced Interior Gateway Routing Protocol)
- OSPF (Open Shortest Path First)

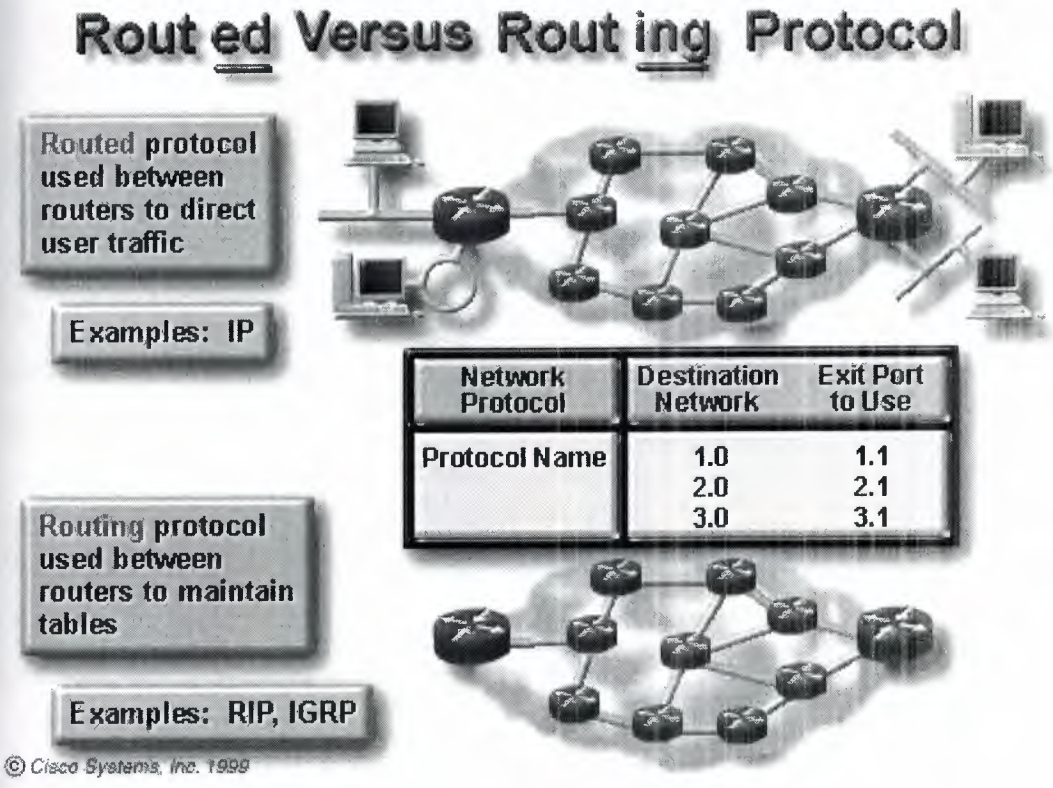

Figure 1-5 Routed Versus Routing Protocol

CHAPTER TWO

Why Routing Protocols Are Necessary

2.1 Static Versus Dynamics Routes

Static route knowledge is administered manually by a network administrator who enters it into a router's configuration. The administrator must manually update this static route entry whenever an inter network topology change requires an update.

Dynamic route knowledge works differently. After a network administrator enters configuration commands to start dynamic routing, the route knowledge is automatically updated by a routing process whenever new information is received from the inter network. Changes in dynamic knowledge are exchanged between routers as part of the update process.

2.2 Why Use a Static Route

Static routing has several useful applications. Dynamic routing tends to reveal everything known about an inter network, for security reasons, you may want to hide parts of an inter network. Static routing enables you to specify the information you want to reveal about restricted networks.

When a network is accessible by only one path as we shown in figure 2.1 , a static route to the network can be sufficient. This type of network is called a stub network. Configuring static routing to a stub network avoids the overhead of dynamic routing.

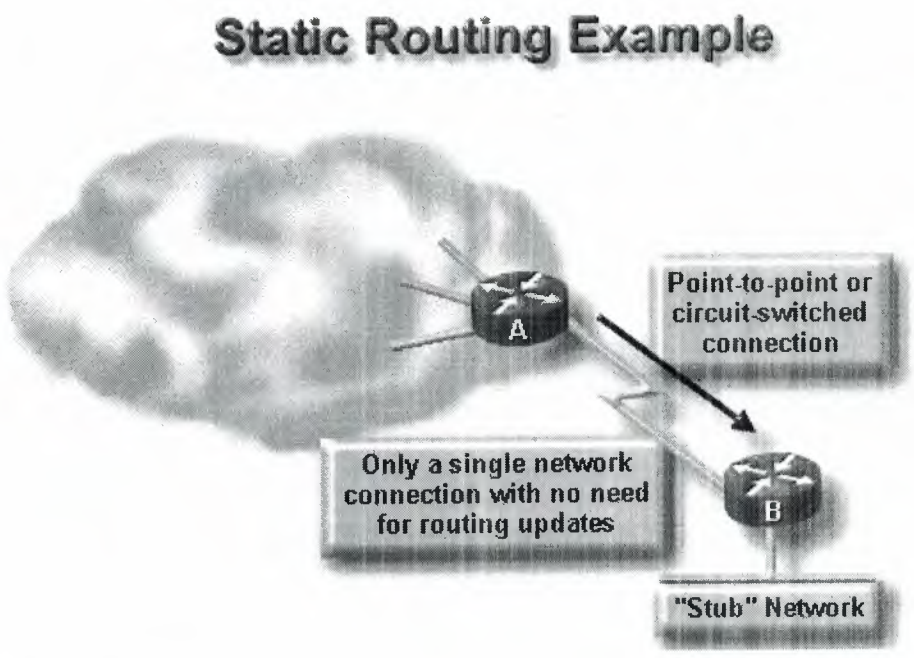

Figure 2-1 Fixed Route To Address Reflects Administrator's Knowledge

2.3 Why Dynamic Routing Is Necessary

The network shown in the Figure adapts differently to topology changes depending on whether it uses statically or dynamically configured routing information.

Static routing allows routers to properly route a packet from network to network based on configured information. The router refers to its routing table and follows the static knowledge residing there to relay the packet to Router D. Router D does the same, and relays the packet to Router C. Router C delivers the packet to the destination host.

If the path between Router A and Router D fails, Router A will not be able to relay the packet to Router D using that static route. Until Router A is manually reconfigured to relay packets by way of Router B, communication with the destination network is impossible.

Dynamic routing offers more flexibility. According to the routing table generated by Router A, a packet can reach its destination over the preferred route through Router D. However, a second path to the destination is available by way of Router B. When Router A recognizes that the link to Router D is down, it adjusts its routing table, making the path through Router B the preferred path to the destination. The routers continue sending packets over this link

When the path between Routers A and D is restored to service, Router A can once again change its routing table to indicate a preference for the counterclockwise path through Routers D and C to the destination network. Dynamic routing protocols can also direct traffic from the same session over different paths in a network for better performance. This is known as loadsharing.

2.4 Dynamic Routing Operations

The success of dynamic routing depends on two basic router functions:

- maintenance of a routing table
- timely distribution of knowledge, in the form of routing updates, to other routers

Dynamic routing relies on a routing protocol to share knowledge among routers as we see in figure 2-2. A routing protocol defines the set of rules used by a router when it communicates with neighboring routers. For example, a routing protocol describes:

- how to send updates
- what knowledge is contained in these updates
- when to send this knowledge
- how to locate recipients of the updates

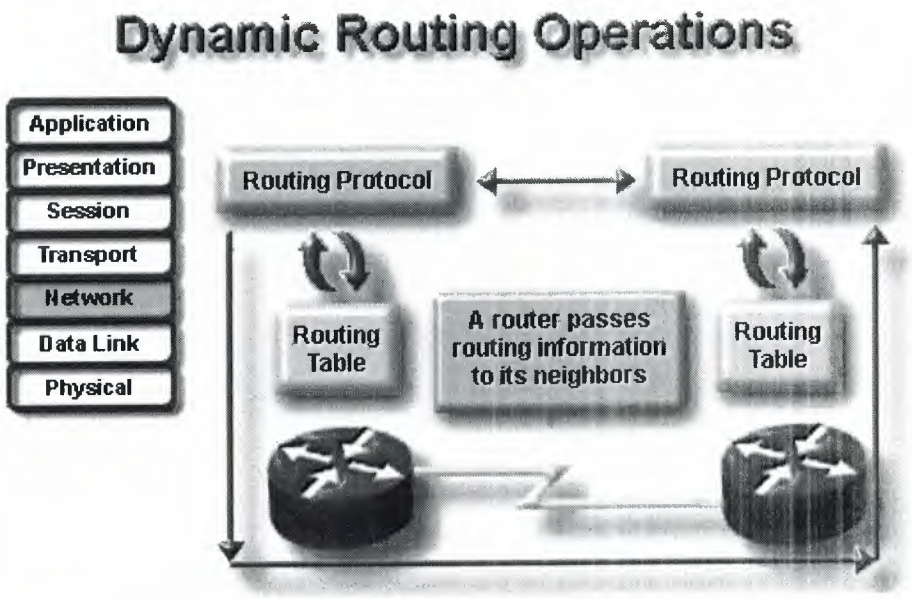

Figure 2.2 Routing Protocol Maintains And Distributes Routing Information

2.5 How Distances On NetWork Paths Are Determined By Various Metrics

When a routing algorithm updates a routing table, its primary objective is to determine the best information to include in the table. Each routing algorithm interprets what is best in its own way. The algorithm generates a number, called the metric value, for each path through the network. Typically, the smaller the metric number, the better the path.

You can calculate metrics based on a single characteristic of a path; you can calculate more complex metrics by combining several characteristics. The metrics most commonly used by routers are as follows:

- bandwidth—the data capacity of a link; (normally, a 10 Mbps Ethernet link is preferable to a 64 kbps leased line)
- delay—the length of time required to move a packet along each link from source to destination

- load—the amount of activity on a network resource such as a router or link
- reliability—usually refers to the error rate of each network link
- hop count—the number of routers a packet must travel through before reaching its destination
- ticks —the delay on a data link using IBM PC clock ticks (approximately 55 milliseconds).
- cost—an arbitrary value, usually based on bandwidth, monetary expense, or other measurement, that is assigned by a network administrator

2.6 Three Classes Of Routing Protocol

Most routing algorithms can be classified as one of two basic algorithms:

- distance vector; or
- link state.

The distance-vector routing approach determines the direction (vector) and distance to any link in the internetwork. The link-state (also called shortest path first) approach re-creates the exact topology of the entire internetwork (or at least the portion in which the router is situated).

The balanced hybrid approach combines aspects of the link-state and distance-vector algorithms. The next several pages cover procedures and problems for each of these routing algorithms and present techniques for minimizing the problems.

2.7 Time To Convergence

The routing algorithm is fundamental to dynamic routing. Whenever the topology of a network changes because of growth, reconfiguration, or failure, the network knowledge base must also change. The knowledge needs to reflect an accurate, consistent view of the new topology. This view is called convergence.

When all routers in an internetwork are operating with the same knowledge, the internetwork is said to have converged. Fast convergence is a desirable network feature because it reduces the period of time in which routers would continue to make incorrect/wasteful routing decisions.

CHAPTER THREE

Distance-Vector Routing

3.1 Distance-Vector Routing Basics

Distance-vector-based routing algorithms pass periodic copies of a routing table from router to router as we see in figure 3.1. These regular updates between routers communicate topology changes.

Each router receives a routing table from its directly connected neighboring routers. For example, in the graphic, Router B receives information from Router A. Router B adds a distance-vector number (such as a number of hops), which increases the distance vector and then passes this new routing table to its other neighbor, Router C. This same step-by-step process occurs in all directions between direct-neighbor routers.

The algorithm eventually accumulates network distances so that it can maintain a database of network topology information. Distance-vector algorithms do not, however, allow a router to know the exact topology of an internetwork.

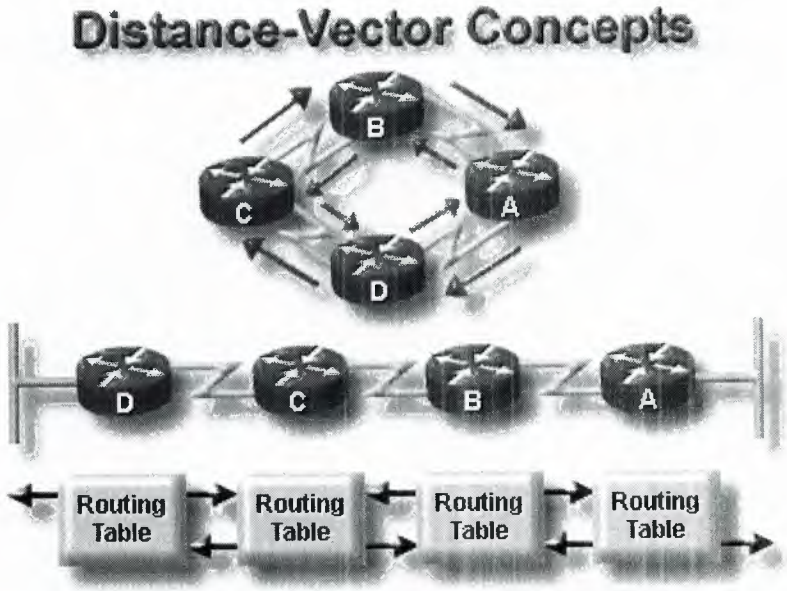

Figure 3.1 Pass Periodic Copies Of Routing Table To Neighbor Routers And Accumulate Distance Vector

3.2 How Distance-Vector Protocols Exchange Routing Tables

Each router that uses distance-vector routing begins by identifying its own neighbors. In the Figure, the interface that leads to each directly-connected network is shown as having a distance of 0. As the distance-vector network discovery process proceeds, routers discover the best path to destination networks based on the information they receive from each neighbor. For example, Router A learns about other networks based on the information that it receives from Router B. Each of the other network entries in the routing table has an accumulated distance vector to show how far away that network is in a given direction.

3.3 How Topology changes Propagate Through The NetWork Of Routers

When the topology in a distance-vector protocol network changes, routing table updates must occur. As with the network discovery process, topology change updates proceed step-by-step from router to router. Distance-vector algorithms call for each router to send its entire routing table to each of its adjacent neighbors. The routing tables include information about the total path cost (defined by its metric) and the logical address of the first router on the path to each network contained in the table.

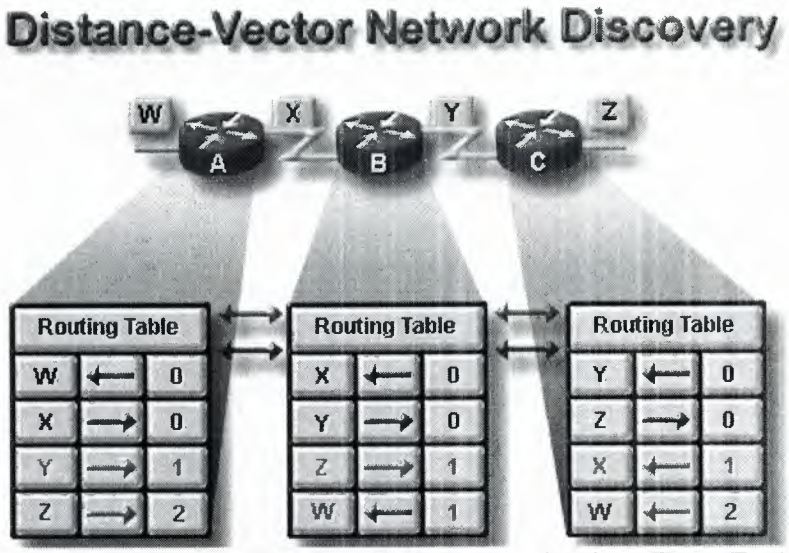

Figure 3.2 Routers Discover The Best Path To Destinations From Each Neighbor

3.4 The Problem Of Routing Loops .

Routing loops can occur if a network's slow convergence on a new configuration causes inconsistent routing entries. The Figure3.3 illustrates how a routing loop can occur:

1. Just before the failure of Network 1, all routers have consistent knowledge and correct routing tables. The network is said to have converged. Assume for the remainder of this example that Router C's preferred path to Network 1 is by way of Router B, and the distance from Router C to Network 1 is 3.

2. When Network 1 fails, Router E sends an update to Router A. Router A stops routing packets to Network 1, but Routers B, C, and D continue to do so because they have not yet been informed of the failure. When Router A sends out its update, Routers B and D stop routing to Network 1; however, Router C has not received an update. To Router C, Network 1 is still reachable via Router B.

3. Now Router C sends a periodic update to Router D, indicating a path to Network 1 by way of Router B. Router D changes its routing table to reflect this good, but incorrect, information, and propagates the information to Router A. Router A propagates the information to Routers B and E, and so on. Any packet destined for Network 1 will now loop from Router C to B to A to D and back to again to C.

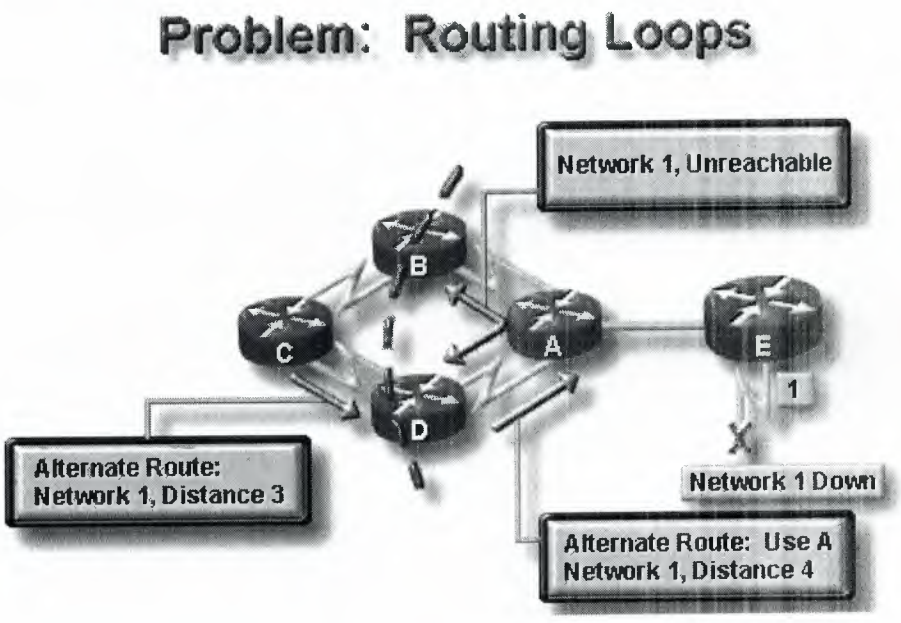

Figure 3.3 Alternative Routes, Slow Convergence, Inconsistent Routing

3.5 The Problem Of Counting To Infinity

Continuing the example from the previous page, the invalid updates of Network 1 will continue to loop until some other process stops the looping. This condition, called count to infinity as shown in figure 3.4, loops packets continuously around the network in spite of the fundamental fact that the destination network, Network 1, is down. While the routers are counting to infinity, the invalid information allows a routing loop to exist.

Without countermeasures to stop the process, the distance vector (metric) of hop count increments each time the packet passes through another router. These packets loop through the network because of wrong information in the routing tables.

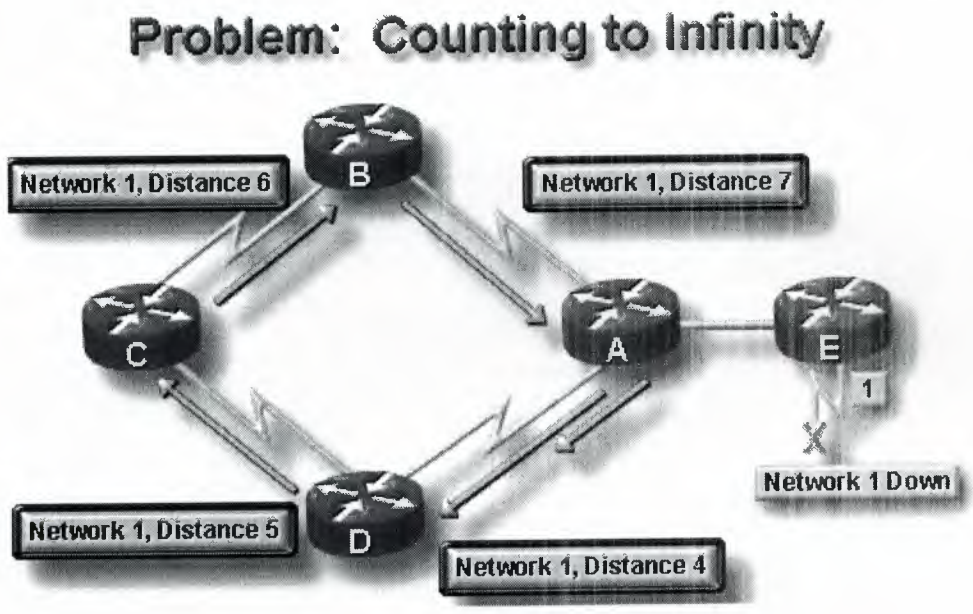

Figure3.4 Routing Loops Increment The Distance Vector

3.6 The Solution Of Defining a Maximum

Distance-vector routing algorithms are self-correcting, but a routing loop problem can require a count to infinity first. To avoid this prolonged problem, distance-vector protocols define infinity as a specific maximum number. This number refers to a routing metric (e.g. a simple hop count).

With this approach, the routing protocol permits the routing loop to continue until the metric exceeds its maximum allowed value. The figure 3.5 shows the metric value as 16 hops, which exceeds the distance-vector default maximum of 15 hops, and the packet is discarded by the router. In any case, when the metric value exceeds the maximum value, Network 1 is considered unreachable.

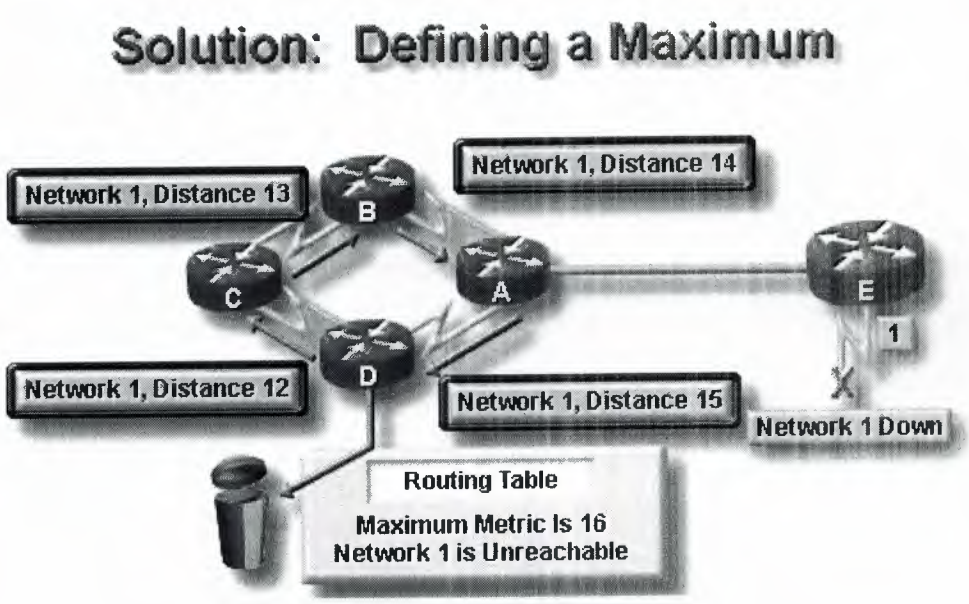

Figure 3.5 Specify a Maximum Distance Vector Metrics As Infinity

3.7 The Solution Of Split Horizon

Another possible source for a routing loop occurs when incorrect information that has been sent back to a router contradicts the correct information that it sent. Here is how this problem occurs:

1. Router A passes an update to Router B and Router D, indicating that Network is down. Router C, however, transmits an update to Router B, indicating that Network 1 is available at a distance of 4, by way of Router D. This does not violate split-horizon rules.

2. Router B concludes, incorrectly, that Router C still has a valid path to Network 1, although at a much less favorable metric. Router B sends an update to Router A advising Router A of the new route to Network 1.

3. Router A now determines that it can send to Network 1 by way of Router B; Router B determines that it can send to Network 1 by way of Router C; and Router C determines that it can send to Network 1 by way of Router D. Any packet introduced into this environment will loop between routers.

4. Split-horizon attempts to avoid this situation. As shown in the Figure 3.6 , if a routing update about Network 1 arrives from Router A, Router B or Router D cannot send information about Network 1 back to Router A. Split-horizon thus reduces incorrect routing information and reduces routing overhead.

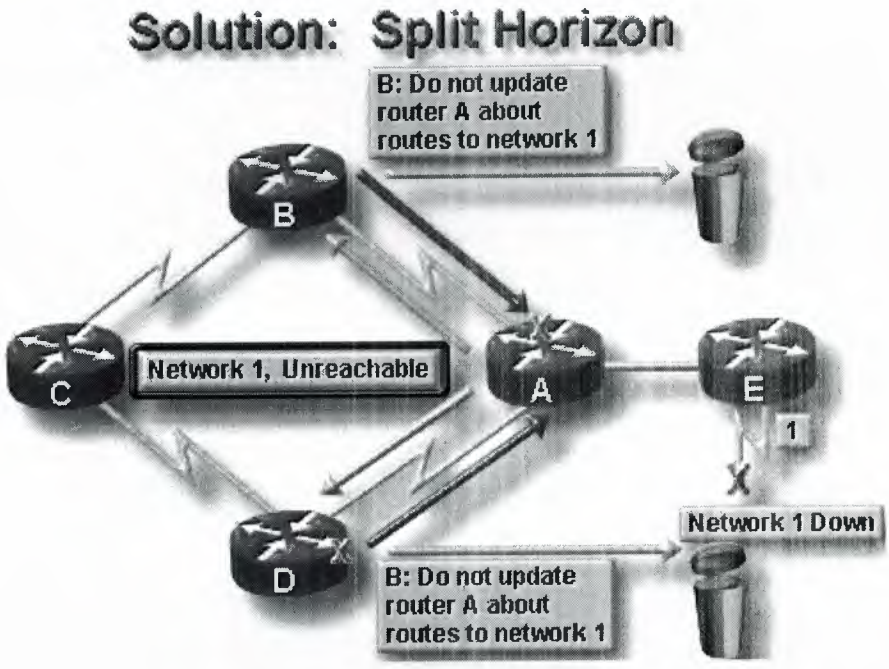

Figure 3.6 If You Learn A protocols Route On An Interface, Do Not Send Information About That Router Back Out That Interface

3.8 The Solution Of Hold-Down Timers

You can avoid a count to infinity problem by using hold-down timers that work as follows:

1. When a router receives an update from a neighbor indicating that a previously accessible network is now inaccessible, the router marks the route as inaccessible and starts a hold-down timer. If at any time before the hold-down timer expires an update is received from the same neighbor indicating that the network is again accessible, the router marks the network as accessible and removes the hold-down timer as we see in figure 3.7.

2. If an update arrives from a different neighboring router with a better metric than originally recorded for the network, the router marks the network as accessible and removes the hold-down timer.

3. If at any time before the hold-down timer expires an update is received from a different neighboring router with a poorer metric, the update is ignored. Ignoring an update with a poorer metric when a hold-down timer is in effect allows more time for the knowledge of a disruptive change to propagate through the entire network.

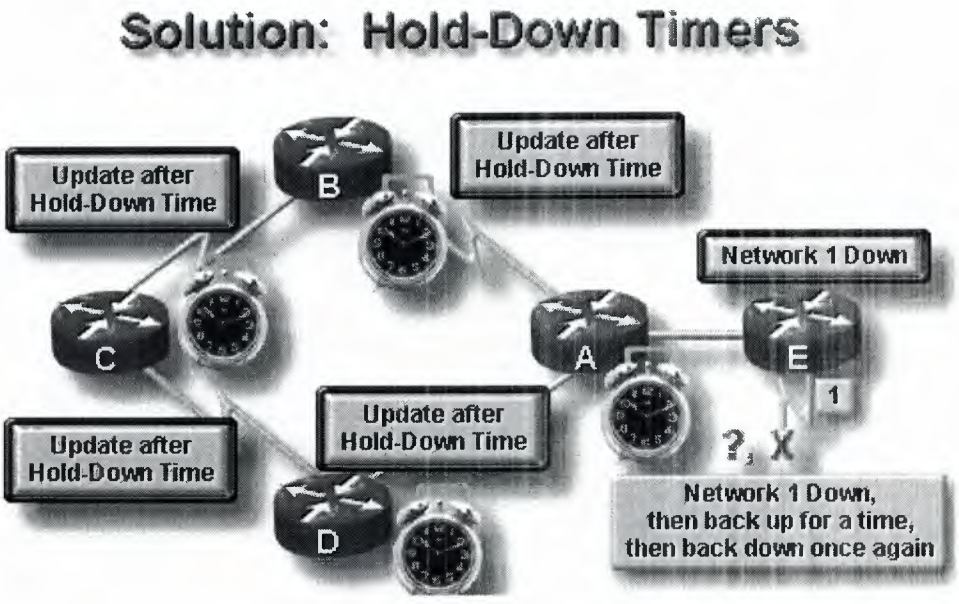

Figure3.7 Router Ignore Network Update Information For Some Period

CHAPTER FOUR

Link State Routing

4.1 Key Characteristics

The second basic algorithm used for routing is the link-state algorithm. Link-state based routing algorithms, also known as SPF (shortest path first) algorithms, maintain a complex database of topology information. Whereas the distance-vector algorithm has nonspecific information about distant networks and no knowledge of distant routers, a link-state routing algorithm maintains full knowledge of distant routers and how they interconnect. Link-state routing uses as shown in figure 4.1:

- link-state advertisements (LSAs)
- a topological database
- the SPF algorithm, and the resulting SPF tree
- a routing table of paths and ports to each network

Engineers have implemented this link-state concept in OSPF (Open Shortest Path First) routing. RFC 1583 contains a description of OSPF link-state concepts and operations.

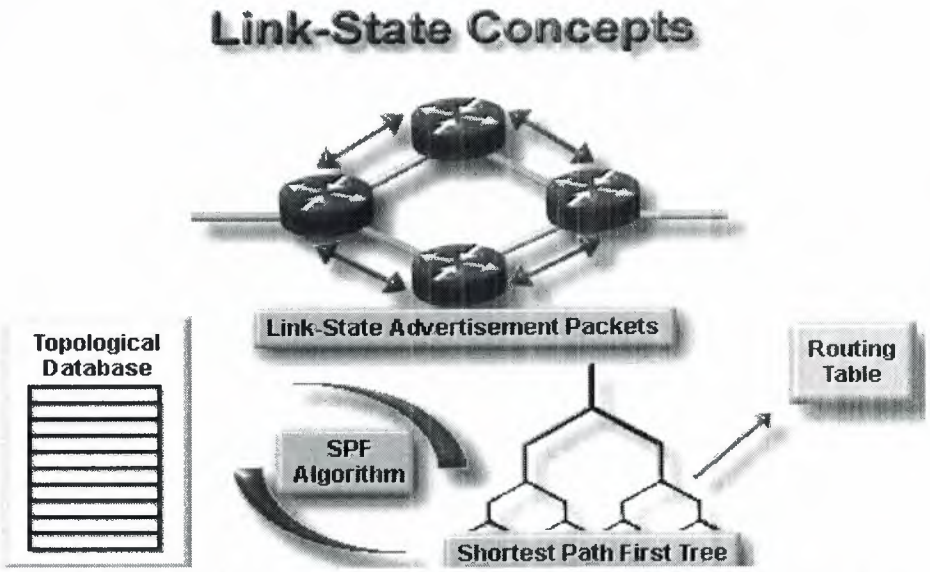

Figure4.1 After Initial flood, pass Small Event-triggered Link-State Updates To All Other Router

4.2 How Link-State Protocols Exchange Routing Tables

Network discovery for link-state routing uses the following processes:

1. Routers exchange LSAs with each other. Each router begins with directly connected networks for which it has direct information.

2. Each router in parallel with the others constructs a topological database consisting of all the LSAs from the internetwork.

3. The SPF algorithm computes network reachability. The router constructs this logical topology as a tree, with itself as root, consisting of all possible paths to each network in the link-state protocol internetwork. It then sorts these paths shortest path first (SPF).

4. The router lists its best paths, and the ports to these destination networks, in the routing table. It also maintains other databases of topology elements and status details.

4.3 How Topology Changes Propagate Through the Network of Routers

Link-state algorithms rely on using the same link-state updates. Whenever a link-state topology changes , the routers that first become aware of the change send information to other routers or to a designated router that all other routers can use for updates.

This involves sending common routing information to all routers in the internetwork. To achieve convergence, each router does the following:

- keeps track of its neighbors: each neighbor's name, whether the neighbor is up or down, and the cost of the link to the neighbor.
- constructs an LSA packet that lists its neighbor router names and link costs, including new neighbors, changes in link costs, and links to neighbors that have gone down.
- sends out this LSA packet so that all other routers receive it.
- when it receives an LSA packet, records the LSA packet in its database so that it updates the most recently generated LSA packet from each router.
- completes a map of the internetwork by using accumulated LSA packet data and then computes routes to all other networks by using the SPF algorithm.

Each time an LSA packet causes a change to the link-state database, the link-state algorithm (SPF) recalculates the best paths and updates the routing table. Then, every router takes the topology change into account as it determines the shortest path to use for packet routing.

4.4 Tow Link-Sate Concerns

There are two link-state concerns - processing and memory requirements, and bandwidth requirements.

4.4.1 Processing and memory requirements

Running link-state routing protocols in most situations requires that routers use more memory and perform more processing than distance-vector routing protocols. Network administrators must ensure that the routers they select are capable of providing these necessary resources

Routers keep track of all other routers in a group and the networks that they can each reach directly. For link-state routing, their memory must be able to hold information from various databases, the topology tree, and the routing table. Using Dijkstra's algorithm to compute the SPF requires a processing task proportional to the number of links in the internetwork, multiplied by the number of routers in the internetwork.

4.4.2 Bandwidth requirements

Another cause for concern involves the bandwidth that must be consumed for initial link-state packet flooding as we see in figure 4.3. During the initial discovery process, all routers using link-state routing protocols send LSA packets to all other routers.

This action floods the internetwork as routers make their en masse demand for bandwidth, and temporarily reduce the bandwidth available for routed traffic that carries user data. After this initial flooding, link-state routing protocols generally require only minimal bandwidth to send infrequent or event-triggered LSA packets that reflect topology changes.

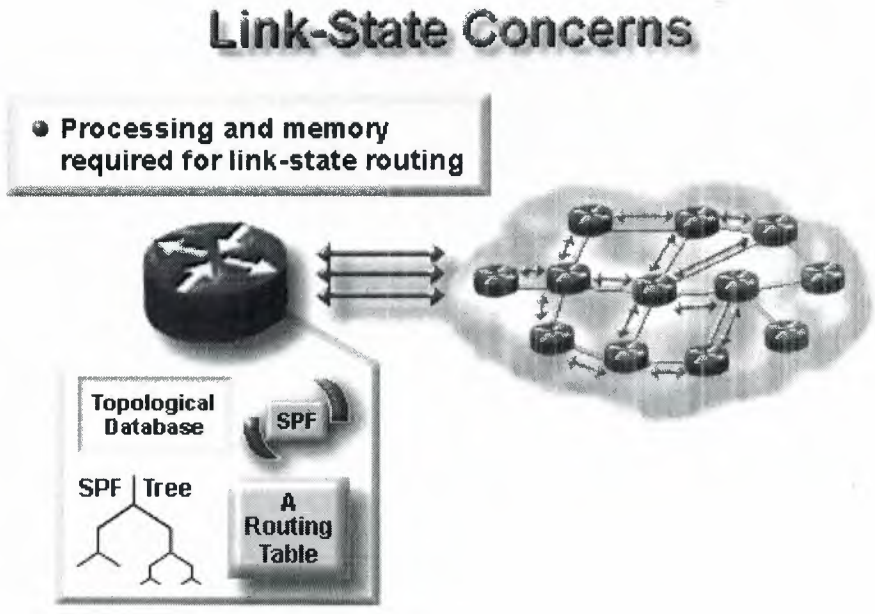

Figure 4.3 Bandwidth Consumed For Initial Link-State (Flood)

4.5 Unsynchronized Link-State Advertisements (LSAs) Leading

The most complex and important aspect of link-state routing is making sure that all routers get all necessary LSA packets. Routers with different sets of LSAs calculate routes based on different topological data. Then, networks become unreachable as a result of a disagreement

among routers about a link. Following is an example of inconsistent path information as we demonstrate in figure 4.4:

1. Between Routers C and D, Network 1 goes down. Both routers construct an LSA packet to reflect this unreachable status.
2. Soon afterward, Network 1 comes back up; another LSA packet reflecting this next topology change is needed.
3. If the original "Network 1, Unreachable" message from Router C uses a slow path for its update, that update comes later. This LSA packet can arrive at Router A after Router D's "Network 1, Back Up Now" LSA.
4. With unsynchronized LSAs, Router A can face a dilemma about which SPF tree to construct. Should it use paths that include Network 1, or paths without Network 1, which was most recently reported as unreachable?

If LSA distribution to all routers is not done correctly, link-state routing can result in invalid routes. Scaling up with link-state protocols on very large internetworks can expand the problem of faulty LSA packet distribution. If one part of the network comes up first with other parts coming up later, the order for sending and receiving LSA packets will vary.

This variation can alter and impair convergence. Routers might learn about different versions of the topology before they construct their SPF trees and routing tables. On a large internetwork, parts that update more quickly can cause problems for parts that update more slowly.

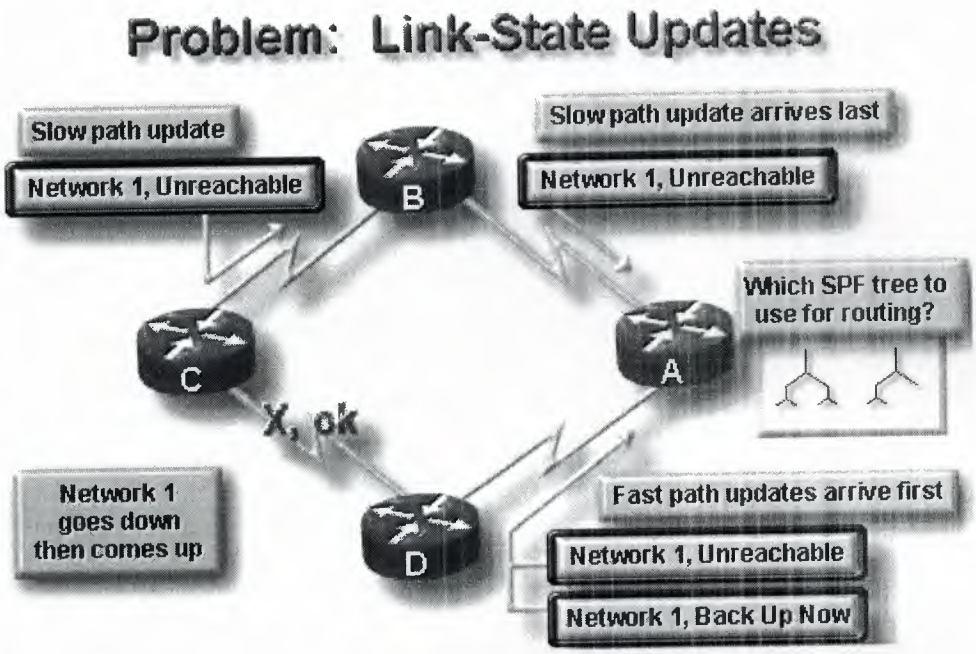

Figure 4.4 Unsynchronized Updates , Inconsistent Path Decision

CHAPTER FIVE

The Context of Different Routing Protocols

5.1 Distance-Vector versus Link-State Routing Protocols

You can compare distance-vector routing to link-state routing in several key areas:

Distance-vector routing gets topological data from the routing table information of its neighbors. Link-state routing obtains a wide view of the entire inter network topology by accumulating all necessary LSAs.

Distance-vector routing determines the best path by adding to the metric value that it receives as routing information is passed from router to router. For link-state routing, each router works separately to calculate its own shortest path to destination networks.

With most distance-vector routing protocols, updates for topology changes come in periodic table updates. The information passes from router to router, usually resulting in slower convergence. With link-state routing protocols, updates are usually triggered by topology changes. Relatively small LSAs passed to all other routers usually result in faster time to converge on any internetwork topology change.

5.2 Hybrid Routing Protocols

An emerging third type of routing protocol combines aspects of both distance-vector and link-state routing. This third type is called balanced-hybrid routing as shown in figure 5.1. Balanced-hybrid routing protocols use distance vectors with more accurate metrics to determine the best paths to destination networks. However, they differ from most distance-vector protocols by using topology changes to trigger routing database updates.

The balanced-hybrid routing protocol converges rapidly, like the link-state protocols. However, it differs from distance-vector and link-state protocols by using fewer resources such as bandwidth, memory, and processor overhead. Examples of hybrid protocols are OSI's IS-IS (Intermediate System-to-Intermediate System), and Cisco's EIGRP (Enhanced Interior Gateway Routing Protocol).

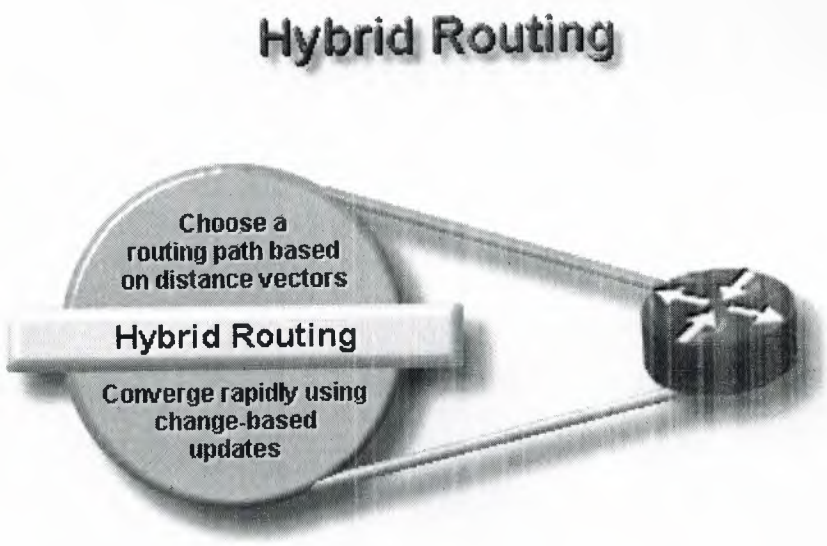

Figure 5.1 Share Attributes Of Both Distance-Vector And Link-State Routing

5.3 LAN-to-LAN Routing

The network layer must understand and be able to interface with various lower layers. Routers must be capable of seamlessly handling packets encapsulated into various lower-level frames without changing the packets' Layer 3 addressing.

The Figure shows an example of this with LAN-to-LAN routing. In this example, packet traffic from source Host 4 on Ethernet Network 1 needs a path to destination Host 5 on Network 2. The LAN hosts depend on the router and its consistent network addressing to find the best path.

When the router checks its routing table entries, it discovers that the best path to destination Network 2 uses outgoing port To0, the interface to a token-ring LAN. Although the lower-layer framing must change as the router passes packet traffic from Ethernet on Network 1 to token-ring on Network 2, the Layer 3 addressing for source and destination remains the same. In the Figure, the destination address remains Network 2, Host 5, regardless of the different lower-layer encapsulations.

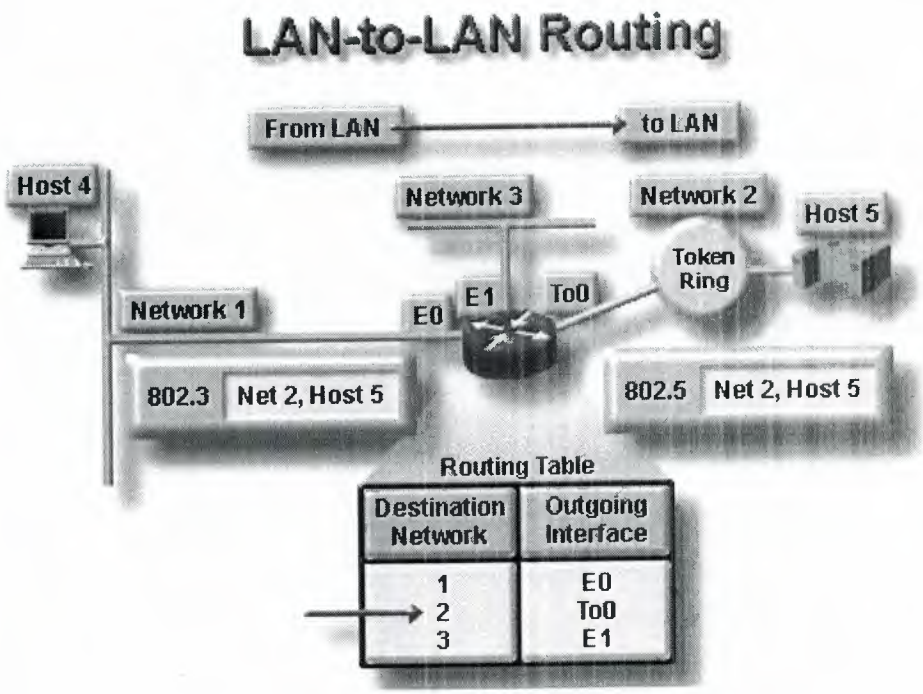

Figure 5.2 Example for LAN-to-LAN routing

5.4 LAN-to-WAN Routing

The network layer must relate to, and interface with, various lower layers for LAN-to-WAN traffic. As an internetwork grows, the path taken by a packet may encounter several relay points and a variety of data link types beyond the LANs. For example, in the Figure 5.3, the following takes place:

1. A packet from the top workstation at address 1.3 must traverse three data links to reach the file server at address 2.4, shown on the bottom.
2. The workstation sends a packet to the file server by first encapsulating it in a token-ring frame addressed to Router A.
3. When Router A receives the frame, it removes the packet from the token-ring frame, encapsulates it in a Frame Relay frame, and forwards the frame to Router B.
4. Router B removes the packet from the Frame Relay frame and forwards it to the file server in a newly created Ethernet frame.

Routers enable LAN-to-WAN packet flow by keeping the end-to-end source and destination addresses constant while encapsulating the packet in data link frames, as appropriate, for the next hop along the path.

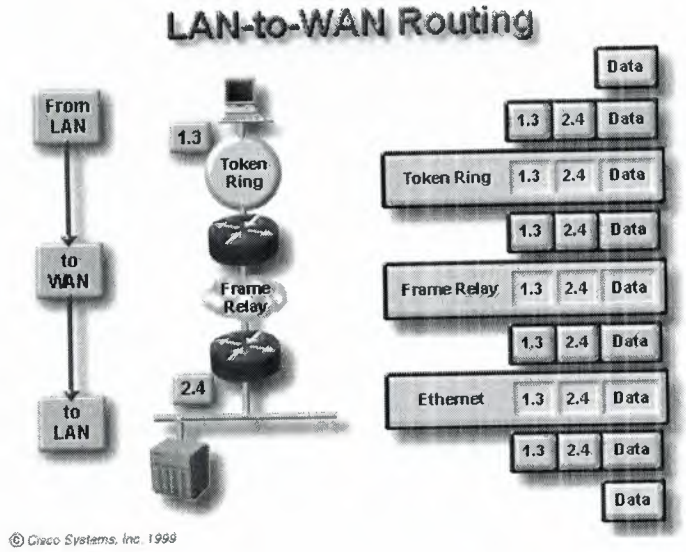

Figure 5.3 example for LAN-to-WAN routing

5.5 Path Selections and Switching of Multiple Protocols and Media

Routers are devices that implement the network service. They provide interfaces for a wide range of links and sub networks at a wide range of speeds. Routers are active and intelligent network nodes that can participate in managing a network. Routers manage networks by providing dynamic control over resources and supporting the tasks and goals for inter network connectivity, reliable performance, management control, and flexibility.

In addition to the basic switching and routing functions, routers have a variety of additional features that help to improve the cost-effectiveness of the internetwork. These features include sequencing traffic based on priority and traffic filtering.

Typically, routers are required to support multiple protocol stacks, each with its own routing protocols, and to allow these different environments to operate in parallel. In practice, routers also incorporate bridging functions and sometimes serve as a limited form of hub.

CHAPTER SIX

Initial Router Configuration

6.1 Setup Mode

After testing the hardware and loading the Cisco IOS system image, the router finds and applies the configuration statements. These entries provide the router with details about router-specific attributes, protocol functions, and interface addresses. However, if the router is unable to locate a valid startup-config file, it enters an initial router configuration mode called setup mode.

With the setup mode command facility, you can answer questions in the system configuration dialog. This facility prompts you for basic configuration information. The answers you enter allow the router to use a sufficient, but minimal-feature, router configuration that includes the following:

- an inventory of interfaces
- an opportunity to enter global parameters
- an opportunity to enter interface parameters
- a setup script review
- an opportunity to indicate whether you want the router to use this configuration

After you approve setup mode entries, the router uses the entries as a running configuration. The router also stores the configuration in NVRAM as a new startup-config, and you can start using the router. For additional protocol and interface changes, you can use the enable mode and enter the command configure.

6.2 Initial Routing Table

Initially, a router must refer to entries about networks or subnets that are directly connected to it. Each interface must be configured with an IP address and a mask. The Cisco IOS software learns about this IP address and mask information from a configuration that has been input from some source. The initial source of addressing is a user who types it into a configuration file.

In the lab that follows, you will start up your router in a just-received condition, a state that lacks another source for the startup configuration. This condition on the router will permit you to use the setup-mode command facility and answer prompts for basic configuration information. The answers you enter will include address-to-port commands to set up router interfaces for IP.

6.3 How a Router Learns about Destinations

By default, routers learn paths to destinations three different ways

- static routes—manually defined by the system administrator as the next hop to a destination; useful for security and traffic reduction
- default routes—manually defined by the system administrator as the path to take when there is no known route to the destination
- dynamic routing—the router learns of paths to destinations by receiving periodic updates from other routers.

6.4 The IP route Command

The ip route command sets up a static route.

The administrative distance is a rating of the trustworthiness of a routing information source, expressed as a numeric value from 0 to 255. The higher the number, the lower the trustworthiness rating.

A static route allows manual configuration of the routing table. No dynamic changes to this table entry will occur as long as the path is active. A static route may reflect some special knowledge of the networking situation known to the network administrator. Manually-entered administrative distance values for static routes are usually low numbers (1 is the default). Routing updates are not sent on a link if they are only defined by a static route, therefore, they conserve bandwidth.

6.5 Using the ip route Command

The assignment of a static route to reach the stub network 172.16.1.0 is proper for Cisco A because there is only one way to reach that network. The assignment of a static route from Cisco B to the cloud networks is also possible. However, a static route assignment is required for each destination network, in which case a default route may be more appropriate.

6.6 The ip default-network Command

The ip default-network command establishes a default route in networks using dynamic routing protocols.

Default routes keep routing tables shorter. When an entry for a destination network does not exist in a routing table, the packet is sent to the default network. Because a router does not have complete knowledge about all destination networks, it can use a default network number to indicate the direction to take for unknown network numbers. Use the default network

number when you need to locate a route but have only partial information about the destination network. The ip default-network command must be added to all routers in the network or used with the additional command redistribute static so all networks have knowledge of the candidate default network.

6.7 Using the ip default-network Command

In the example, the global command ip default-network 192.168.17.0 defines the Class C network 192.168.17.0 as the destination path for packets that have no routing table entries. The Company X administrator does not want updates coming in from the public network. Router A could need a firewall for routing updates. Router A may need a mechanism to group those networks that will share Company X's routing strategy. One such mechanism is an autonomous system number.

CHAPTER SEVEN

Interior Gateway Routing Protocol (IGRP)

7.1 IGRP

The Interior Gateway Routing Protocol (IGRP) is a routing protocol that was developed in the mid-1980s by Cisco Systems, Inc. Cisco's principal goal in creating IGRP was to provide a robust protocol for routing within an autonomous system (AS) as shown in figure 7.1.

In the mid-1980s, the most popular intra-AS routing protocol was the Routing Information Protocol (RIP). Although RIP was quite useful for routing within small-to moderate-sized, relatively homogeneous internetworks, its limits were being pushed by network growth. In particular, RIP's small hop-count limit (16) restricted the size of internetworks, and its single metric (hop count) did not allow for much routing flexibility in complex environments. The popularity of Cisco routers and the robustness of IGRP have encouraged many organizations with large internetworks to replace RIP with IGRP.

Cisco's initial IGRP implementation worked in Internet Protocol (IP) networks. IGRP was designed to run in any network environment, however, and Cisco soon ported it to run in OSI Connectionless-Network Protocol (CLNP) networks. Cisco developed Enhanced IGRP in the early 1990s to improve the operating efficiency of IGRP. This chapter discusses IGRP's basic design and implementation. Enhanced IGRP is discussed in "Enhanced IGRP."

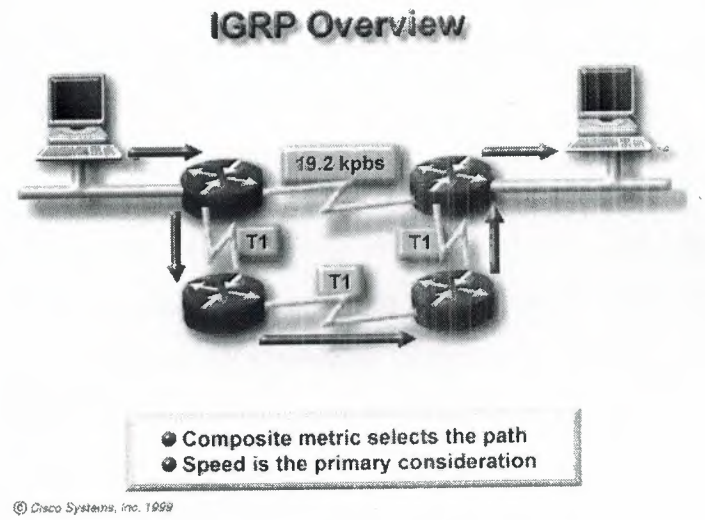

Figure 7.1 IGRP Protocol example

7.2 IGRP Protocol Characteristics

IGRP is a distance-vector interior gateway protocol (IGP). Distance-vector routing protocols call for each router to send all or a portion of its routing table in a routing-update message at regular intervals to each of its neighboring routers. As routing information proliferates through the network, routers can calculate distances to all nodes within the internetwork.

IGRP uses a combination (vector) of metrics. Internetwork delay, bandwidth, reliability, and load are all factored into the routing decision. Network administrators can set the weighting factors for each of these metrics. IGRP uses either the administrator-set or the default weightings to automatically calculate optimal routes.

IGRP provides a wide range for its metrics. Reliability and load, for example, can take on any value between 1 and 255; bandwidth can take on values reflecting speeds from 1,200 bps to 10 gigabits per second, while delay can take on any value from 1 to 2 to the 24th power. Wide metric ranges allow satisfactory metric setting in internetworks with widely varying

performance characteristics. Most importantly, the metric components are combined in a user-definable algorithm. As a result, network administrators can influence route selection in an intuitive fashion.

To provide additional flexibility, IGRP permits multipath routing. Dual equal-bandwidth lines can run a single stream of traffic in round-robin fashion, with automatic switchover to the second line if one line goes down. Also, multiple paths can be used even if the metrics for the paths are different. If, for example, one path is three times better than another because its metric is three times lower, the better path will be used three times as often. Only routes with metrics that are within a certain range of the best route are used as multiple paths.

7.3 Stability Features

IGRP provides a number of features that are designed to enhance its stability. These include hold-downs, split horizons, and poison-reverse updates.

Hold-downs are used to prevent regular update messages from inappropriately reinstating a route that might have gone bad. When a router goes down, neighboring routers detect this via the lack of regularly scheduled update messages. These routers then calculate new routes and send routing update messages to inform their neighbors of the route change. This activity begins a wave of triggered updates that filter through the network. These triggered updates do not instantly arrive at every network device, so it is therefore possible for a device that has yet to be informed of a network failure to send a regular update message (indicating that a route that has just gone down is still good) to a device that has just been notified of the network failure. In this case, the latter device would contain (and potentially advertise) incorrect routing information. Hold-downs tell routers to hold down any changes that might affect routes for some period of time. The hold-down period usually is calculated to be just greater than the period of time necessary to update the entire network with a routing change.

Split horizons derive from the premise that it is never useful to send information about a route back in the direction from which it came. Figure 7.2 illustrates the split-horizon rule. Router 1 (R1) initially advertises that it has a route to Network A. There is no reason for Router 2 (R2) to include this route in its update back to R1 because R1 is closer to Network A. The split-horizon rule says that R2 should strike this route from any updates it sends to R1. The split-horizon rule helps prevent routing loops. Consider, for example, the case where R1's interface to Network A goes down. to R1horizons, R2 continues to inform R1 that it can get to Network A (through R1). If R1 does not have sufficient intelligence, it actually might pick up R2's route as an alternative to its failed direct connection, causing a routing loop. Although hold-downs should prevent this, split horizons are implemented in IGRP because they provide extra algorithm stability.

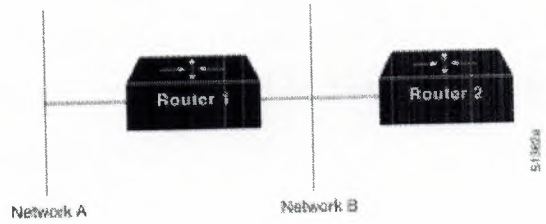

Figure 7.2 the split horizons rule helps protect against routing loops.

Split horizons should prevent routing loops between adjacent routers, but poison-reverse updates are necessary to defeat larger routing loops. Increases in routing metrics generally indicate routing loops. Poison-reverse updates then are sent to remove the route and place it in hold-down. In Cisco's implementation of IGRP, poison-reverse updates are sent if a route metric has increased by a factor of 1.1 or greater.

7.4 Timers

IGRP maintains a number of timers and variables containing time intervals. These include an update timer, an invalid timer, a hold-time period, and a flush timer.

The update timer specifies how frequently routing update messages should be sent. The IGRP default for this variable is 90 seconds. The invalid timer specifies how long a router should wait, in the absence of routing-update messages about a specific route before declaring that route invalid. The IGRP default for this variable is three times the update period.

The hold-time variable specifies the hold-down period. The IGRP default for this variable is three times the update timer period plus 10 seconds. Finally, the flush timer indicates how much time should pass before a route should be flushed from the routing table. The IGRP default is seven times the routing update period.

7.5 Key Characteristics of IGRP

IGRP is a distance-vector routing protocol developed by Cisco. IGRP sends routing updates at 90 second intervals, advertising networks for a particular autonomous system. Some of the IGRP key design characteristics emphasize the following:

- versatility that enables it to automatically handle indefinite, complex topologies
- flexibility for segments that have different bandwidth and delay characteristics
- scalability for functioning in very large networks

The IGRP routing protocol by default uses two metrics, bandwidth and delay. IGRP can be configured to use a combination of variables to determine a composite metric. Those variables include:

- bandwidth
- delay
- load
- reliability

CHAPTER EIGHT

Open Shortest Path First (OSPF)

8.1 OSPF

Open Shortest Path First (OSPF) is a routing protocol developed for Internet Protocol (IP) networks by the interior gateway protocol (IGP) working group of the Internet Engineering Task Force (IETF). The working group was formed in 1988 to design an IGP based on the shortest path first (SPF) algorithm for use in the Internet. Similar to the Interior Gateway Routing Protocol (IGRP), OSPF was created because in the mid-1980s, the Routing Information Protocol (RIP) was increasingly unable to serve large, heterogeneous internetworks. This chapter examines the OSPF routing environment, underlying routing algorithm and general protocol components

.

OSPF was derived from several research efforts, including Bolt, Beranek, Newman's (BBN's) SPF algorithm developed in 1978 for the ARPANET (a landmark packet-switching network developed in the early 1970s by BBN), Dr. Radia Perlman's research on fault-tolerant broadcasting of routing information (1988), BBN's work on area routing (1986), and an early version of OSI's Intermediate System-to-Intermediate System (IS-IS) routing protocol.

OSPF has two primary characteristics. The first is that the protocol is open, which means that its specification is in the public domain. The OSPF specification is published as Request For Comments (RFC) 1247. The second principal characteristic is that OSPF is based on the SPF algorithm, which sometimes is referred to as the Dijkstra algorithm, named for the person credited with its creation.

OSPF is a link-state routing protocol that calls for the sending of link-state advertisements (LSAs) to all other routers within the same hierarchical area. Information on attached

interfaces, metrics used, and other variables is included in OSPF LSAs. As OSPF routers accumulate link-state information, they use the SPF algorithm to calculate the shortest path to each node.

As a link-state routing protocol, OSPF contrasts with RIP and IGRP, which are distance-vector routing protocols. Routers running the distance-vector algorithm send all or a portion of their routing tables in routing-update messages to their neighbors.

8.2 Routing Hierarchy

Unlike RIP, OSPF can operate within a hierarchy. The largest entity within the hierarchy is the autonomous system (AS), which is a collection of networks under a common administration that share a common routing strategy. OSPF is an intra-AS (interior gateway) routing protocol, although it is capable of receiving routes from and sending routes to other ASs.

An AS can be divided into a number of areas, which are groups of contiguous networks and attached hosts. Routers with multiple interfaces can participate in multiple areas. These routers, which are called area border routers, maintain separate topological databases for each area.

A topological database is essentially an overall picture of networks in relationship to routers. The topological database contains the collection of LSAs received from all routers in the same area. Because routers within the same area share the same information, they have identical topological databases.

The term domain sometimes is used to describe a portion of the network in which all routers have identical topological databases. Domain is frequently used interchangeably with AS.

An area's topology is invisible to entities outside the area. By keeping area topologies separate, OSPF passes less routing traffic than it would if the AS were not partitioned.

Area partitioning creates two different types of OSPF routing, depending on whether the source and destination are in the same or different areas. Intra-area routing occurs when the source and destination are in the same area; interarea routing occurs when they are in different areas.

An OSPF backbone is responsible for distributing routing information between areas. It consists of all area border routers, networks not wholly contained in any area, and their attached routers. Figure 8.1 shows an example of an internetwork with several areas.

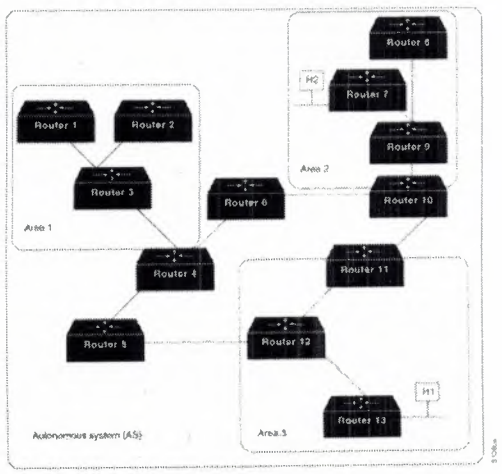

Figure 8.1: An OSPF AS consists of multiple areas linked by routers In the figure

Routers 4, 5, 6, 10, 11, and 12 make up the backbone. If Host H1 in Area 3 wants to send a packet to Host H2 in area 2, the packet is sent to Router 13, which forwards the packet to Router 12, which sends the packet to Router 11. Router 11 then forwards the packet along the backbone to area border Router 10, which sends the packet through two intra-area routers (Router 9 and Router 7) to be forwarded to Host H2.

The backbone itself is an OSPF area, so all backbone routers use the same procedures and algorithms to maintain routing information within the backbone that any area router would. The backbone topology is invisible to all intra-area routers, as are individual area topologies to the backbone.

Areas can be defined in such a way that the backbone is not contiguous. In this case, backbone connectivity must be restored through virtual links. Virtual links are configured between any backbone routers that share a link to a nonbackbone area and function as if they were direct links.

AS border routers running OSPF learn about exterior routes through exterior gateway protocols (EGPs), such as Exterior Gateway Protocol (EGP) or Border Gateway Protocol (BGP), or through configuration information.

8.3 SPF Algorithm

The shortest path first (SPF) routing algorithm is the basis for OSPF operations. When an SPF router is powered up, it initializes its routing-protocol data structures and then waits for indications from lower-layer protocols that its interfaces are functional.

After a router is assured that its interfaces are functioning, it uses the OSPF Hello protocol to acquire neighbors, which are routers with interfaces to a common network. The router sends hello packets to its neighbors and receives their hello packets. In addition to helping acquire neighbors, hello packets also act as keep-alives to let routers know that other routers are still functional.

On multiaccess networks (networks supporting more than two routers), the Hello protocol elects a designated router and a backup designated router. Among other things, the designated router is responsible for generating LSAs for the entire multiaccess network. Designated routers allow a reduction in network traffic and in the size of the topological database.

When the link-state databases of two neighboring routers are synchronized, the routers are said to be adjacent. On multiaccess networks, the designated router determines which routers should become adjacent. Topological databases are synchronized between pairs of adjacent routers. Adjacencies control the distribution of routing-protocol packets, which are sent and received only on adjacencies.

Each router periodically sends an LSA to provide information on a router's adjacencies or to inform others when a router's state changes. By comparing established adjacencies to link states, failed routers can be detected quickly and the network's topology altered appropriately. From the topological database generated from LSAs, each router calculates a shortest-path tree, with itself as root. The shortest-path tree, in turn, yields a routing table.

8.4 Packet Format

All OSPF packets begin with a 24-byte header, as illustrated in Figure 8.2 .

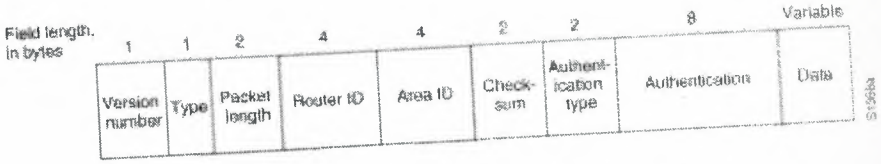

Figure 8.2 OSPF packets consist of nine fields.

The following descriptions summarize the header fields illustrated in figure 8.2.

- Version Number---Identifies the OSPF version used.

- Type---Identifies the OSPF packet type as one of the following:
    - Hello: Establishes and maintains neighbor relationships.
    - Database Description: Describes the contents of the topological database. These messages are exchanged when an adjacency is initialized.
    - Link-state Request: Requests pieces of the topological database from neighbor routers. These messages are exchanged after a router discovers (by examining database-description packets) that parts of its topological database are out of date.
    - Link-state Update: Responds to a link-state request packet. These messages also are used for the regular dispersal of LSAs. Several LSAs can be included within a single link-state update packet.
    - Link-state Acknowledgment: Acknowledges link-state update packets.
- Packet Length---Specifies the packet length, including the OSPF header, in bytes.
- Router ID---Identifies the source of the packet.
- Area ID---Identifies the area to which the packet belongs. All OSPF packets are associated with a single area.
- Checksum---Checks the entire packet contents for any damage suffered in transit.
- Authentication Type---Contains the authentication type. All OSPF protocol exchanges are authenticated. The Authentication Type is configurable on a per-area basis.
- Authentication---Contains authentication information.

Data---Contains encapsulated upper-layer information.

8.5 Additional OSPF Features

Additional OSPF features include equal-cost, multipath routing, and routing based on upper-layer type-of-service (TOS) requests. TOS-based routing supports those upper-layer protocols that can specify particular types of service. An application, for example, might specify that certain data is urgent. If OSPF has high-priority links at its disposal, these can be used to transport the urgent datagram.

OSPF supports one or more metrics. If only one metric is used, it is considered to be arbitrary, and TOS is not supported. If more than one metric is used, TOS is optionally supported through the use of a separate metric (and, therefore, a separate routing table) for each of the eight combinations created by the three IP TOS bits (the delay, throughput, and reliability bits). If, for example, the IP TOS bits specify low delay, low throughput, and high reliability, OSPF calculates routes to all destinations based on this TOS designation.

IP subnet masks are included with each advertised destination, enabling variable-length subnet masks. With variable-length subnet masks, an IP network can be broken into many subnets of various sizes. This provides network administrators with extra network-configuration flexibility.

CHAPTER NINE

Enhanced Interior Gateway Routing Protocol (EIGRP)

9.1 EIGRP

The Enhanced Interior Gateway Routing Protocol (IGRP) represents an evolution from its predecessor IGRP This evolution resulted from changes in networking and the demands of diverse, large-scale internetworks. Enhanced IGRP integrates the capabilities of link-state protocols into distance-vector protocols. It incorporates the Diffusing-Update Algorithm (DUAL) developed at SRI International by Dr. J.J. Garcia-Luna-Aceves.

Enhanced IGRP provides compatibility and seamless interoperation with IGRP routers. An automatic-redistribution mechanism allows IGRP routes to be imported into Enhanced IGRP, and vice versa, so it is possible to add Enhanced IGRP gradually into an existing IGRP network. Because the metrics for both protocols are directly translatable, they are as easily comparable as if they were routes that originated in their own Autonomous Systems (ASs). In addition, Enhanced IGRP treats IGRP routes as external routes and provides a way for the network administrator to customize them.

This chapter provides an overview of the basic operations and protocol characteristics of Enhanced IGRP.

9.2 Enhanced IGRP Capabilities and Attributes

Key capabilities that distinguish Enhanced IGRP from other routing protocols include fast convergence, support variable-length subnet mask, support for partial updates, and support for multiple network-layer protocols.

A router running Enhanced IGRP stores all its neighbors' routing tables so that it can quickly adapt to alternate routes. If no appropriate route exists, Enhanced IGRP queries its neighbors to discover an alternate route. These queries propagate until an alternate route is found.

Its support for variable-length subnet masks permits routes to be automatically summarized on a network number boundary. In addition, Enhanced IGRP can be configured to summarize on any bit boundary at any interface.

Enhanced IGRP does not make periodic updates. Instead, it sends partial updates only when the metric for a route changes. Propagation of partial updates is automatically bounded so that only those routers that need the information are updated. As a result of these two capabilities, Enhanced IGRP consumes significantly less bandwidth than IGRP.

Enhanced IGRP includes support for AppleTalk, IP, and Novell NetWare. The AppleTalk implementation redistributes routes learned from the Routing Table Maintenance Protocol (RTMP). The IP implementation redistributes routes learned from OSPF, Routing Information Protocol (RIP), IS-IS, Exterior Gateway Protocol (EGP), or Border Gateway Protocol (BGP). The Novell implementation redistributes routes learned from Novell RIP or Service Advertisement Protocol (SAP).

9.3 Underlying Processes and Technologies

To provide superior routing performance, Enhanced IGRP employs four key technologies that combine to differentiate it from other routing technologies: neighbor discovery/recovery, reliable transport protocol (RTP), DUAL finite-state machine, and protocol-dependent modules.

Neighbor discovery/recovery is used by routers to dynamically learn about other routers on their directly attached networks. Routers also must discover when their neighbors become

unreachable or inoperative. This process is achieved with low overhead by periodically sending small hello packets. As long as a router receives hello packets from a neighboring router, it assumes that the neighbor is functioning, and the two can exchange routing information.

Reliable Transport Protocol (RTP) is responsible for guaranteed, ordered delivery of Enhanced IGRP packets to all neighbors. It supports intermixed transmission of multicast or unicast packets. For efficiency, only certain Enhanced IGRP packets are transmitted reliably. On a multiaccess network that has multicast capabilities, such as Ethernet, it is not necessary to send hello packets reliably to all neighbors individually. For that reason, Enhanced IGRP sends a single multicast hello packet containing an indicator that informs the receivers that the packet need not be acknowledged. Other types of packets, such as updates, indicate in the packet that acknowledgment is required. RTP contains a provision for sending multicast packets quickly when unacknowledged packets are pending, which helps ensure that convergence time remains low in the presence of varying speed links.

DUAL finite-state machine embodies the decision process for all route computations by tracking all routes advertised by all neighbors. DUAL uses distance information to select efficient, loop-free paths and selects routes for insertion in a routing table based on feasible successors. A feasible successor is a neighboring router used for packet forwarding that is a least-cost path to a destination that is guaranteed not to be part of a routing loop. When a neighbor changes a metric, or when a topology change occurs, DUAL tests for feasible successors. If one is found, DUAL uses it to avoid recomputing the route unnecessarily.

When no feasible successors exist but neighbors still advertise the destination, a recomputation (also known as a diffusing computation) must occur to determine a new successor. Although recomputation is not processor-intensive, it does affect convergence time, so it is advantageous to avoid unnecessary recomputations.

Protocol-dependent modules are responsible for network-layer protocol-specific requirements. The IP-Enhanced IGRP module, for example, is responsible for sending and receiving Enhanced IGRP packets that are encapsulated in IP. Likewise, IP-Enhanced IGRP is also responsible for parsing Enhanced IGRP packets and informing DUAL of the new information that has been received.

IP-Enhanced IGRP asks DUAL to make routing decisions, the results of which are stored in the IP routing table. IP-Enhanced IGRP is responsible for redistributing routes learned by other IP routing protocols.

9.4 Neighbor Tables

When a router discovers a new neighbor, it records the neighbor's address and interface as an entry in the *neighbor table*. One neighbor table exists for each protocol-dependent module. When a neighbor sends a hello packet, it advertises a hold time, which is the amount of time a router treats a neighbor as reachable and operational. If a hello packet is not received within the hold time, the hold time expires and DUAL is informed of the topology change.

The neighbor-table entry also includes information required by RTP. Sequence numbers are employed to match acknowledgments with data packets, and the last sequence number received from the neighbor is recorded so that out-of-order packets can be detected. A transmission list is used to queue packets for possible retransmission on a per-neighbor basis. Round-trip timers are kept in the neighbor-table entry to estimate an optimal retransmission interval.

9.5 Topology Tables

The topology table contains all destinations advertised by neighboring routers. The protocol-dependent modules populate the table, and the table is acted on by the DUAL finite-state machine. Each entry in the topology table includes the destination address and a list of neighbors that have advertised the destination. For each neighbor, the entry records the

advertised metric, which the neighbor stores in its routing table. An important rule that distance vector protocols must follow is that if the neighbor advertises this destination, it must use the route to forward packets.

The metric that the router uses to reach the destination is also associated with the destination. The metric that the router uses in the routing table, and to advertise to other routers, is the sum of the best advertised metric from all neighbors, plus the link cost to the best neighbor.

9.6 Route States

A topology-table entry for a destination can exist in one of two states: active or passive. A destination is in the passive state when the router is not performing a recomputation, or in the active state when the router is performing a recomputation. If feasible successors are always available, a destination never has to go into the active state, thereby avoiding a recomputation.

A recomputation occurs when a destination has no feasible successors. The router initiates the recomputation by sending a query packet to each of its neighboring routers. The neighboring router can send a reply packet, indicating it has a feasible successor for the destination, or it can send a query packet, indicating that it is participating in the recomputation. While a destination is in the active state, a router cannot change the destination's routing-table information. After the router has received a reply from each neighboring router, the topology-table entry for the destination returns to the passive state, and the router can select a successor.

9.7 Route Tagging

Enhanced IGRP supports internal and external routes. Internal routes originate within an Enhanced IGRP AS. Therefore, a directly attached network that is configured to run Enhanced IGRP is considered an internal route and is propagated with this information throughout the Enhanced IGRP AS. External routes are learned by another routing protocol or reside in the

routing table as static routes. These routes are tagged individually with the identity of their origin.

External routes are tagged with the following information:

- Router ID of the Enhanced IGRP router that redistributed the route
- AS number of the destination
- Configurable administrator tag
- ID of the external protocol
- Metric from the external protocol
- Bit flags for default routing

Route tagging allows the network administrator to customize routing and maintain flexible policy controls. Route tagging is particularly useful in transit ASs,

where Enhanced IGRP typically interacts with an interdomain routing protocol that implements more global policies, resulting in a very scalable, policy-based routing.

9.8 Enhanced IGRP Packet Types

Enhanced IGRP uses the following packet types: hello and acknowledgment, update, and query and reply.

Hello packets are multicast for neighbor discovery/recovery and do not require acknowledgment. An acknowledgment packet is a hello packet that has no data. Acknowledgment packets contain a non-zero acknowledgment number and always are sent by using a unicast address.

Update packets are used to convey reachability of destinations. When a new neighbor is discovered, unicast update packets are sent so that the neighbor can build up its topology table.

In other cases, such as a link-cost change, updates are multicast. Updates always are transmitted reliably.

Query and reply packets are sent when a destination has no feasible successors. Query packets are always multicast. Reply packets are sent in response to query packets to instruct the originator not to recompute the route because feasible successors exist. Reply packets are unicast to the originator of the query. Both query and reply packets are transmitted reliably.

CHAPTER TEN

Border Gateway Protocol (BGP)

10.1 BGP

Routing involves two basic activities: determination of optimal routing paths and the transport of information groups (typically called packets) through an internetwork. The transport of packets through an internetwork is relatively straightforward. Path determination, on the other hand, can be very complex. One protocol that addresses the task of path determination in today's networks is the Border Gateway Protocol (BGP). This chapter summarizes the basic operations of BGP and provides a description of its protocol components.

BGP performs interdomain routing in Transmission-Control Protocol/Internet Protocol (TCP/IP) networks. BGP is an exterior gateway protocol (EGP), which means that it performs routing between multiple autonomous systems or domains and exchanges routing and reachability information with other BGP systems.

BGP was developed to replace its predecessor, the now obsolete Exterior Gateway Protocol (EGP), as the standard exterior gateway-routing protocol used in the global Internet. BGP solves serious problems with EGP and scales to Internet growth more efficiently; Figure 10.1 illustrates core routers using BGP to route traffic between autonomous systems.

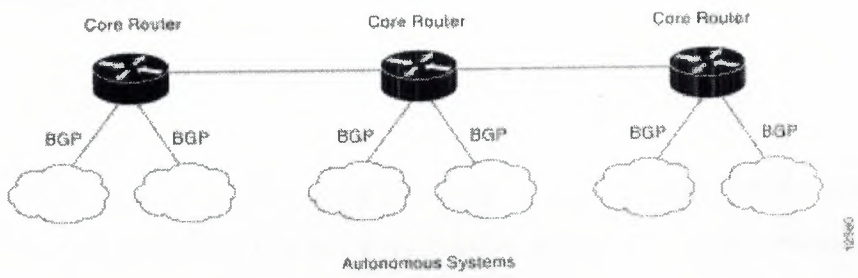

Figure 10.1 Core routers can use BGP to route traffic between autonomous systems.

BGP is specified in several Request For Comments (RFCs)

- RFC 1771 ---Describes BGP4, the current version of BGP
- RFC 1654---Describes the first BGP4 specification

RFC 1105, RFC 1163, and RFC 1267---Describes versions of BGP prior to BGP4

10.2 BGP Operations

BGP performs three types of routing: interautonomous system routing, intra-autonomous system routing, and pass-through autonomous system routing.

Interautonomous system routing occurs between two or more BGP routers in different autonomous systems. Peer routers in these systems use BGP to maintain a consistent view of the internetwork topology. BGP neighbors communicating between autonomous systems must reside on the same physical network.

The Internet serves as an example of an entity that uses this type of routing because it is comprised of autonomous systems or administrative domains. Many of these domains represent the various institutions, corporations, and entities that make up the Internet.

BGP is frequently used to provide path determination to provide optimal routing within the Internet.

Intra-autonomous system routing occurs between two or more BGP routers located within the same autonomous system. Peer routers within the same autonomous system use BGP to maintain a consistent view of the system topology. BGP also is used to determine which router will serve as the connection point for specific external autonomous systems. Once again, the Internet provides an example of interautonomous system routing. An organization, such as a university, could make use of BGP to provide optimal routing within its own administrative domain or autonomous system. The BGP protocol can provide both inter- and intra-autonomous system routing services.

Pass-through autonomous system routing occurs between two or more BGP peer routers that exchange traffic across an autonomous system that does not run BGP. In a pass-through autonomous system environment, the BGP traffic did not originate within the autonomous system in question and is not destined for a node in the autonomous system. BGP must interact with whatever intra-autonomous system routing protocol is being used to successfully transport BGP traffic through that autonomous system. Figure 10.2 illustrates a pass-through autonomous system environment:

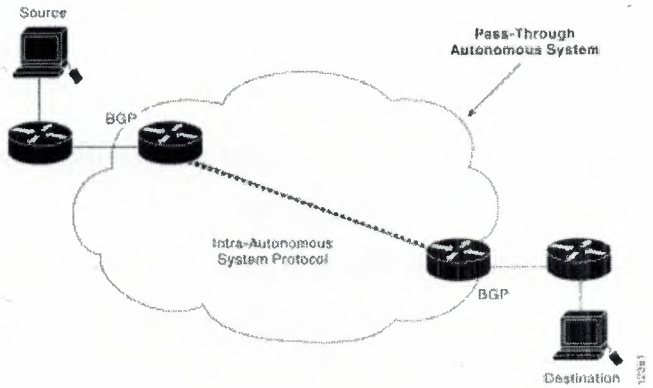

Figure 10.2: In pass-through autonomous system routing, BGP pairs with another intra-autonomous system-routing protocol.

10.3 BGP Routing

As with any routing protocol, BGP maintains routing tables, transmits routing updates, and bases routing decisions on routing metrics. The primary function of a BGP system is to exchange network-reachability information, including information about the list of autonomous system paths, with other BGP systems. This information can be used to construct a graph of autonomous system connectivity from which routing loops can be pruned and with which autonomous system-level policy decisions can be enforced.

Each BGP router maintains a routing table that lists all feasible paths to a particular network. The router does not refresh the routing table, however. Instead, routing information received from peer routers is retained until an incremental update is received.

BGP devices exchange routing information upon initial data exchange and after incremental updates. When a router first connects to the network, BGP routers exchange their entire BGP routing tables. Similarly, when the routing table changes, routers send the portion of their routing table that has changed. BGP routers do not send regularly scheduled routing updates, and BGP routing updates advertise only the optimal path to a network.

BGP uses a single routing metric to determine the best path to a given network. This metric consists of an arbitrary unit number that specifies the degree of preference of a particular link. The BGP metric typically is assigned to each link by the network administrator. The value assigned to a link can be based on any number of criteria, including the number of autonomous systems through which the path passes, stability, speed, delay, or cost.

10.4 BGP Message Types

Four BGP message types are specified in RFC 1771, A Border Gateway Protocol 4 (BGP-4): open message, update message, notification message, and keep-alive message.

The open message opens a BGP communications session between peers and is the first message sent by each side after a transport-protocol connection is established. Open messages are confirmed using a keep-alive message sent by the peer device and must be confirmed before updates, notifications, and keep-alives can be exchanged.

An update message is used to provide routing updates to other BGP systems, allowing routers to construct a consistent view of the network topology. Updates are sent using the Transmission-Control Protocol (TCP) to ensure reliable delivery. Update messages can withdraw one or more unfeasible routes from the routing table and simultaneously can advertise a route while withdrawing others.

The notification message is sent when an error condition is detected. Notifications are used to close an active session and to inform any connected routers of why the session is being closed.

The keep-alive message notifies BGP peers that a device is active. Keep-alives are sent often enough to keep the sessions from expiring.

10.5 BGP Packet Formats

The sections that follow summarize BGP open, updated, notification, and keep-alive message types, as well as the basic BGP header format. Each is illustrated with a format drawing, and the fields shown are defined.

10.6 Header Format

All BGP message types use the basic packet header. Open, update, and notification messages have additional fields, but keep-alive messages use only the basic packet header. Figure 10.3 illustrates the fields used in the BGP header. The section that follows summarizes the function of each field.

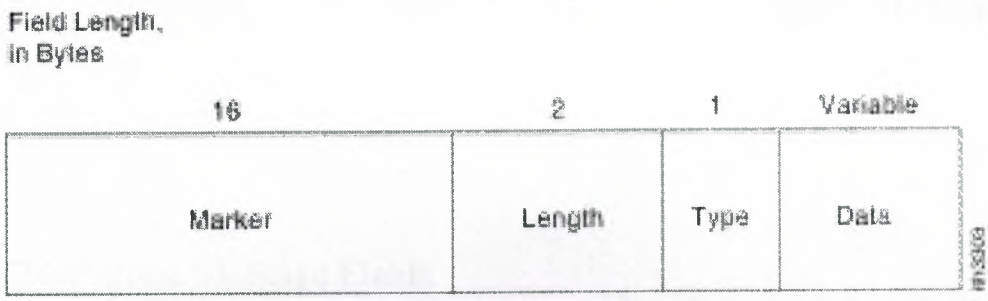

Figure 10.3 A BGP packet header consists of four fields.

10.7 BGP Packet-Header Fields

Each BGP packet contains a header whose primary purpose is to identify the function of the packet in question. The following descriptions summarize the function of each field in the BGP header illustrated in Figure 10.3.

- Marker---Contains an authentication value that the message receiver can predict.
- Length---Indicates the total length of the message in bytes.
- Type---Type --- Specifies the message type as one of the following:
    - Open
    - Update
    - Notification
    - Keep-alive
- Data---Contains upper-layer information in this optional field.

10.8 Open Message Format

BGP open messages are comprised of a BGP header and additional fields. Figure 10.

4 illustrates the additional fields used in BGP open messages.

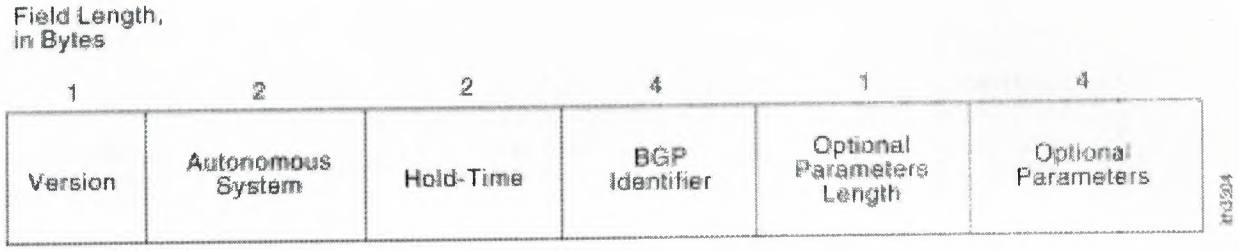

Figure 10.4: A BGP open message consists of six fields.

10.9 BGP Open Message Fields

BGP packets in which the type field in the header identifies the packet to be a BGP open message packet include the following fields. These fields provide the exchange criteria for two BGP routers to establish a peer relationship.

- Version---Provides the BGP version number so that the recipient can determine whether it is running the same version as the sender.
- Autonomous System---Provides the autonomous system number of the sender.
- Hold-Time---Indicates the maximum number of seconds that can elapse without receipt of a message before the transmitter is assumed to be nonfunctional.
- BGP Identifier---Provides the BGP identifier of the sender (an IP address), which is determined at startup and is identical for all local interfaces and all BGP peers.
- Optional Parameters Length---Indicates the length of the optional parameters field (if present).
- Optional Parameters---Contains a list of optional parameters (if any). Only one optional parameter type is currently defined: authentication information.

10.10 Update Message Format

BGP update messages are comprised of a BGP header and additional fields. Figure 10.5 illustrates the additional fields used in BGP update messages.

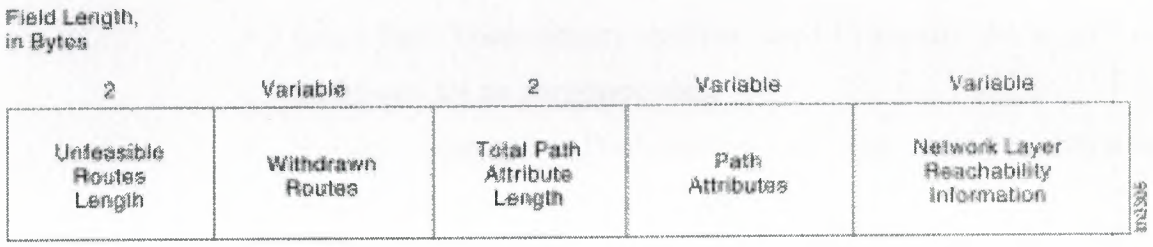

Figure 10.5: A BGP update message contains five fields.

10.11 BGP Update Message Fields

BGP packets in which the type field in the header identifies the packet to be a BGP update message packet include the following fields. Upon receiving an update message packet, routers will be able to add or delete specific entries from their routing tables to ensure accuracy. Update messages consist of the following packets:

- Unfeasible Routes Length---Indicates the total length of the withdrawn routes field or that the field is not present.
- Withdrawn Routes---Contains a list of IP address prefixes for routes being withdrawn from service.
- Total Path Attribute Length---Indicates the total length of the path attributes field or that the field is not present.
- Path Attributes---Describes the characteristics of the advertised path. The following are possible attributes for a path:
    - Origin: Mandatory attribute that defines the origin of the path information
    - AS Path: Mandatory attribute composed of a sequence of autonomous system path segments

- Next Hop: Mandatory attribute that defines the IP address of the border router that should be used as the next hop to destinations listed in the network layer reachability information field
- Mult Exit Disc: Optional attribute used to discriminate between multiple exit points to a neighboring autonomous system
- Local Pref: Discretionary attribute used to specify the degree of preference for an advertised route
- Atomic Aggregate: Discretionary attribute used to disclose information about route selections
- Aggregator: Optional attribute that contains information about aggregate routes

Network Layer Reachability Information---Contains a list of IP address prefixes for the advertised routes

10.12 Notification Message Format

Figure 10.6 illustrates the additional fields used in BGP notification messages.

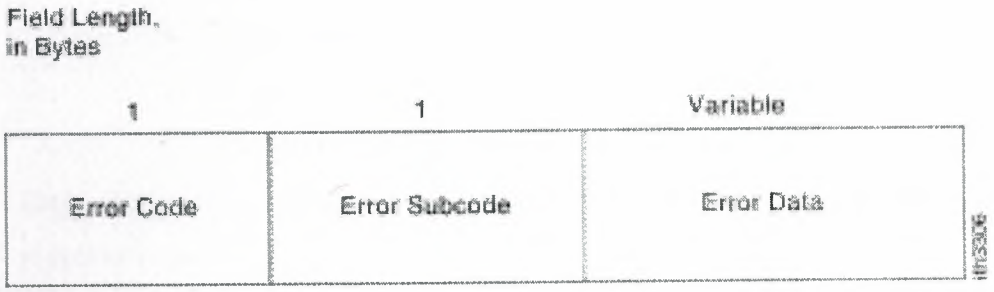

Figure 10.6 A BGP notification message consists of three fields.

10.13 BGP Notification Message Fields

BGP packets in which the type field in the header identifies the packet to be a BGP notification message packet include the following fields. This packet is used to indicate some sort of error condition to the peers of the originating router.

- Error Code---Indicates the type of error that occurred. The following are the error types defined by the field:
    - Message Header Error: Indicates a problem with a message header, such as unacceptable message length, unacceptable marker field value, or unacceptable message type.
    - Open Message Error: Indicates a problem with an open message, such as unsupported version number, unacceptable autonomous system number or IP address, or unsupported authentication code.
    - Update Message Error: Indicates a problem with an update message, such as a malformed attribute list, attribute list error, or invalid next-hop attribute.
    - Hold Time Expired: Indicates that the hold-time has expired, after which time a BGP node will be considered nonfunctional.
    - Finite State Machine Error: Indicates an unexpected event.
    - Cease: Closes a BGP connection at the request of a BGP device in the absence of any fatal errors.
- Error Subcode---Provides more specific information about the nature of the reported error.
- Error Data---Contains data based on the error code and error subcode fields. This field is used to diagnose the reason for the notification message.

CONCLUSION

Routers keep track of all other routers in a group and the networks that they can each reach directly. For link-state routing, their memory must be able to hold information from various databases, the topology tree, and the routing table. Using Dijkstra's algorithm to compute the SPF requires a processing task proportional to the number of links in the internetwork, multiplied by the number of routers in the internetwork.

The primary function of a BGP system is to exchange network reachability information with other BGP systems. This information is used to construct a graph of AS connectivity from which routing loops are pruned and with which AS-level policy decisions are enforced. BGP provides a number of techniques for controlling the flow of BGP updates, such as route, path, and community filtering. It also provides techniques for consolidating routing information, such as CIDR aggregation, confederations, and route reflectors. BGP is a powerful tool for providing loop-free interdomain routing within and between ASs.

REFERENCE

[1] Cisco Systems Inc. Network Designs 1999

[2] Sun Microsystems, Enterprise Services 2002

[3] 3Com Corporation, Understanding Routers 2001

[4] Networking and Network Programming, IDG Books 2003

[5] "TFTP Protocol (Revision 2)," Sollins, K.R.; 1998

[6] Router Protocols and how they work, ITP Publications 2000

[7] James Chellis, Charles Perkings, Matthew Strebe, "Networking Essentials", SYBEX Publishers 1999.