MODULE 3

Networking Protocol Suites

CHAPTER 10

TCP/IP Overview

Objectives

Upon completion of this module, you will be able to perform the following tasks:

Describe how the TCP/IP implementation relates to the OSI reference model

Identify the functions of the TCP/IP transport-layer protocols

Identify the functions of the TCP/IP network-layer protocols

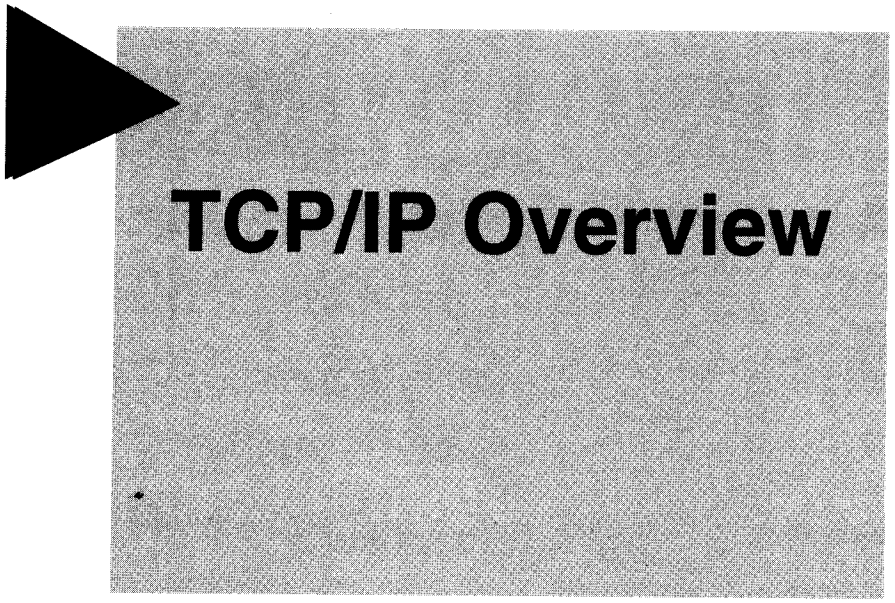
Identify the functions performed by ICMP

2

This chapter provides an overview of TCP/IP. It presents information about TCP and UDP at the transport layer and IP, ICMP, ARP, and RARP at the network layer.

Sections:

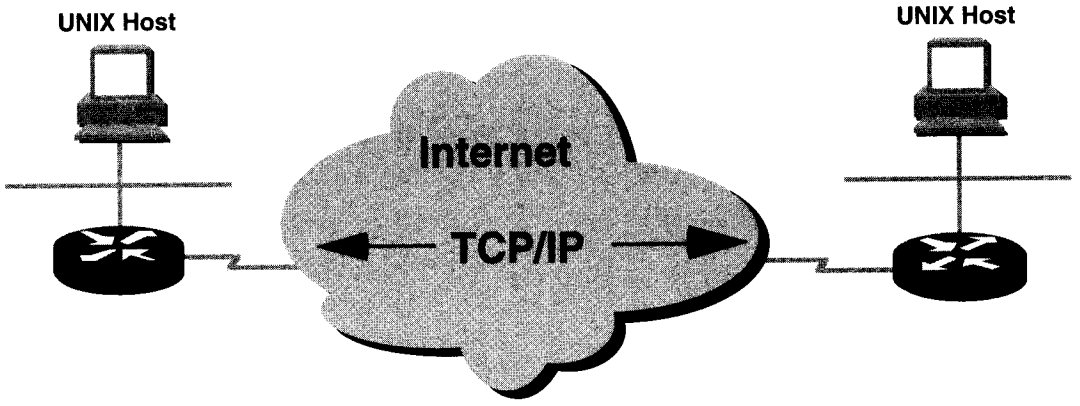
- TCP/IP Overview
- Transport Layer
- Network Layer
- Answers to Exercises



3

TCP/IP Overview

Introduction to TCP/IP



- **Early protocol suite**
- **Universal**

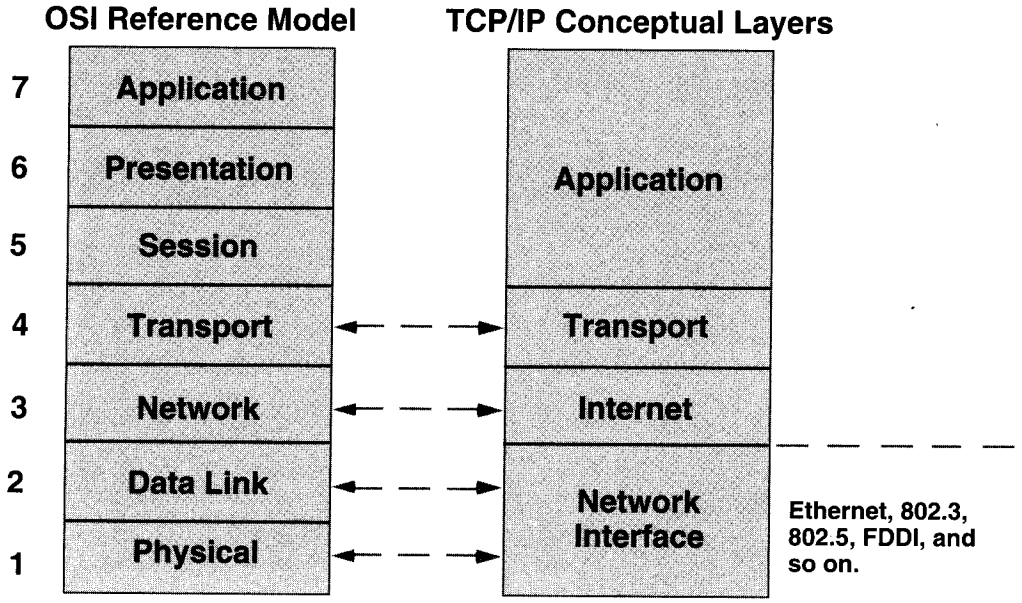
4

The Transmission Control Protocol/Internet Protocol (TCP/IP) suite of protocols was developed as part of the research done by the Defense Advanced Research Projects Agency (DARPA). Later TCP/IP was included with the Berkeley Software Distribution of UNIX.

The internet protocols can be used to communicate across any set of interconnected networks. They are equally well-suited for both LAN and WAN communication.

The internet protocol suite includes not only Layer 3 and 4 specifications (such as IP and TCP), but also specifications for such common applications as e-mail, remote login, terminal emulation, and file transfer.

TCP/IP Protocol Stack

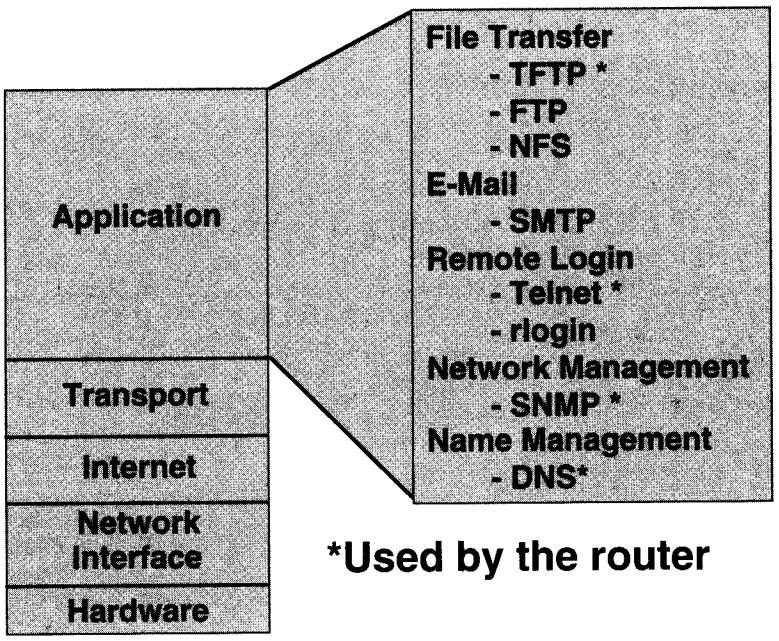


5

The TCP/IP protocol stack maps closely to the OSI reference model in the lower layers. All standard physical and data-link protocols are supported.

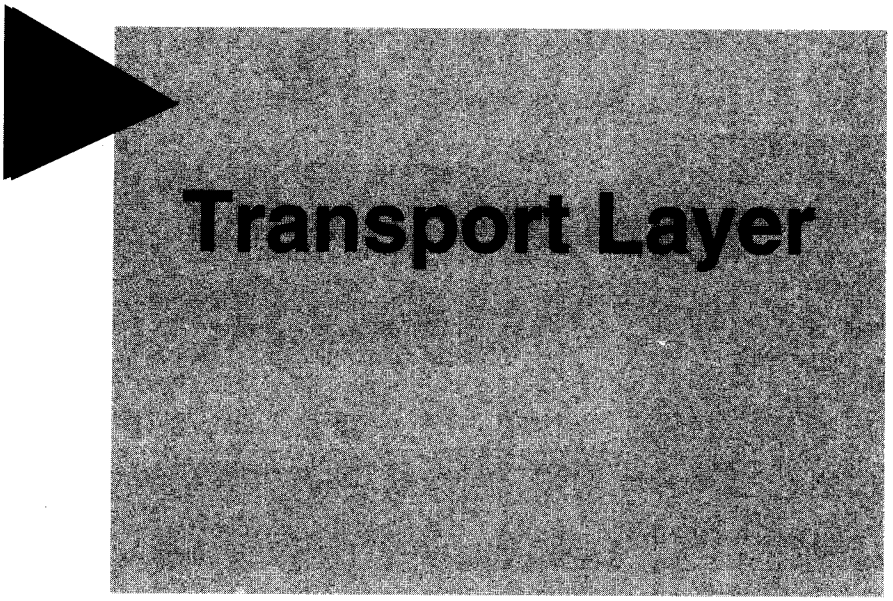
TCP/IP information is transferred in a sequence of datagrams. One message may be transmitted as a series of datagrams that are reassembled into the message at the receiving location.

Application Layer Overview



6

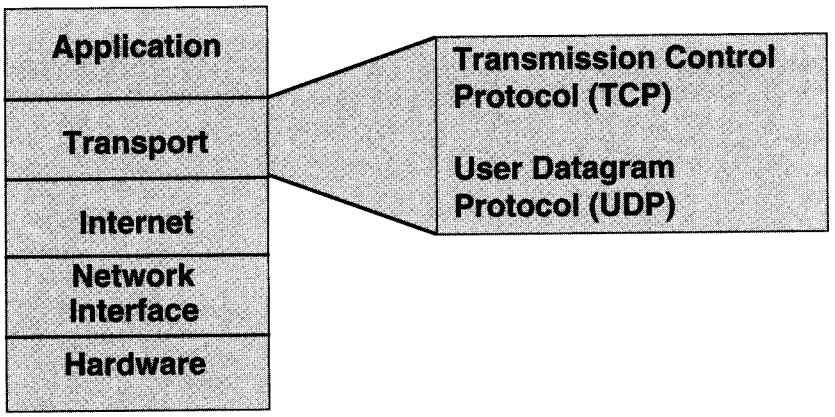
Application protocols exist for file transfer, e-mail, and remote login. Network management is also supported at the application layer.



7

Transport Layer

Transport Layer Overview



8

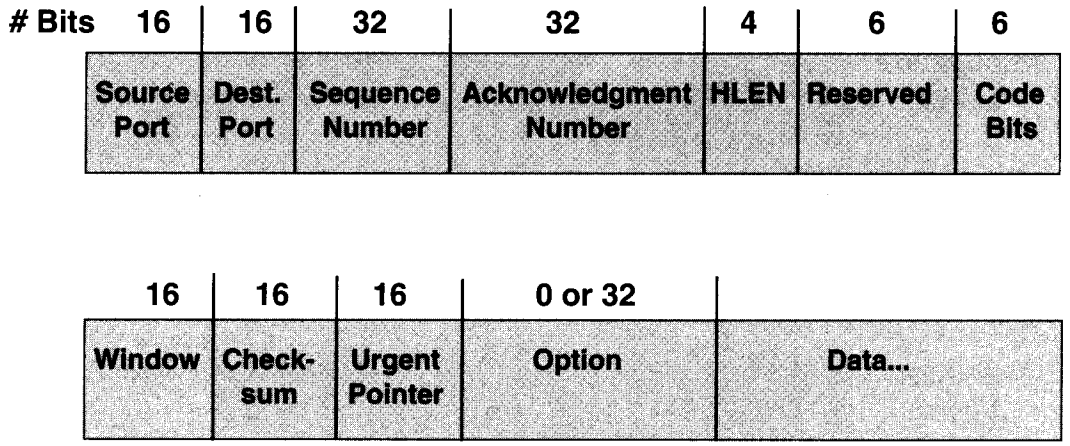
The transport layer performs two functions:

- Flow control provided by sliding windows
- Reliability provided by sequence numbers and acknowledgments

Two protocols are provided at the transport layer: TCP and UDP.

- TCP is a connection-oriented, reliable protocol. It is responsible for breaking messages into segments, reassembling them at the destination station, resending anything that is not received, and reassembling messages from the segments. TCP supplies a virtual circuit between end-user applications.
- UDP is connectionless and "unreliable." Although UDP is responsible for transmitting messages, no software checking for segment delivery is provided at this layer; hence the description "unreliable."

TCP Segment Format

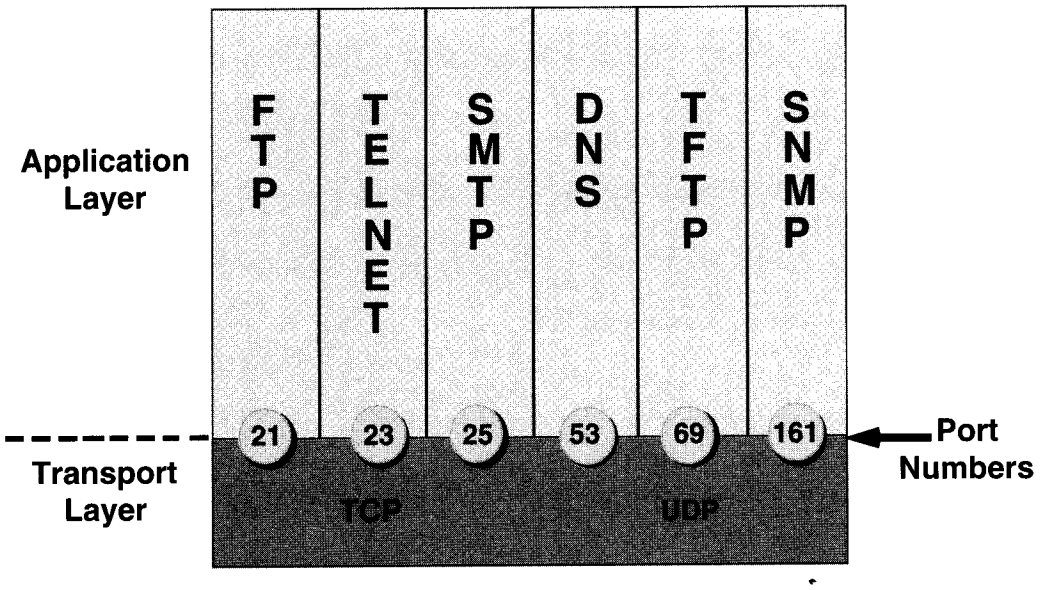


9

Field definitions in the TCP segment:

- Source Port—Number of the calling port
- Destination Port—Number of the called port
- Sequence Number—Number used to ensure correct sequencing of the arriving data
- Acknowledgment Number—Next expected TCP octet
- HLEN—Number of 32-bit words in the header
- Reserved—Set to zero
- Code Bits—Control functions (such as setup and termination of a session)
- Window—Number of octets that the sender is willing to accept
- Checksum—Calculated checksum of the header and data fields
- Urgent Pointer—Indicates the end of the urgent data
- Option—One currently defined: maximum TCP segment size
- Data—Upper-layer protocol data

Port Numbers



10

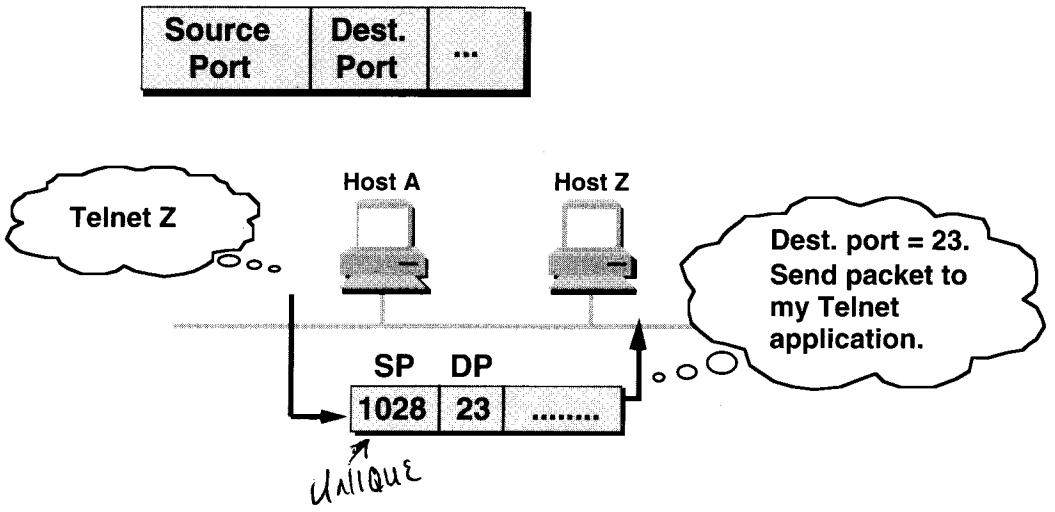
Both TCP and UDP use port (or socket) numbers to pass information to the upper layers. Port numbers are used to keep track of different conversations crossing the network at the same time.

Application software developers agree to use well-known port numbers that are defined in RFC 1700. For example, any conversation bound for the FTP application uses the standard port number 21. Conversations that do not involve an application with a well-known port number are assigned port numbers randomly chosen from within a specific range instead. These port numbers are used as source and destination addresses in the TCP segment.

Some ports are reserved in both TCP and UDP, but applications might not be written to support them. Port numbers have the following assigned ranges:

- Numbers below 255 are for public applications.
- Numbers from 255 to 1023 are assigned to companies for saleable applications.
- Numbers above 1023 are unregulated.

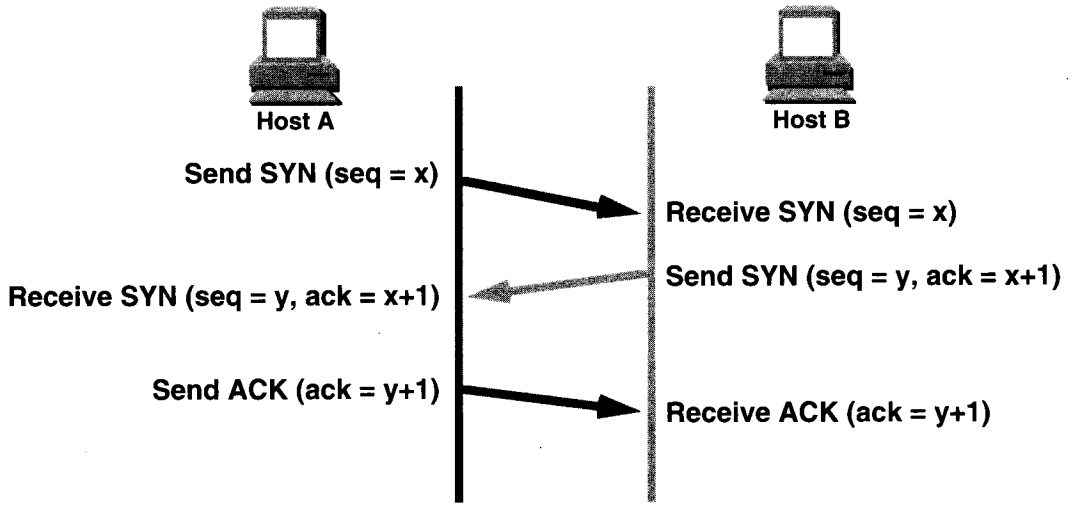
TCP Port Numbers



11

End systems use port numbers to select the proper application. Originating source port numbers are dynamically assigned by the source host, usually some number greater than 1023.

TCP Three-Way Handshake/ Open Connection

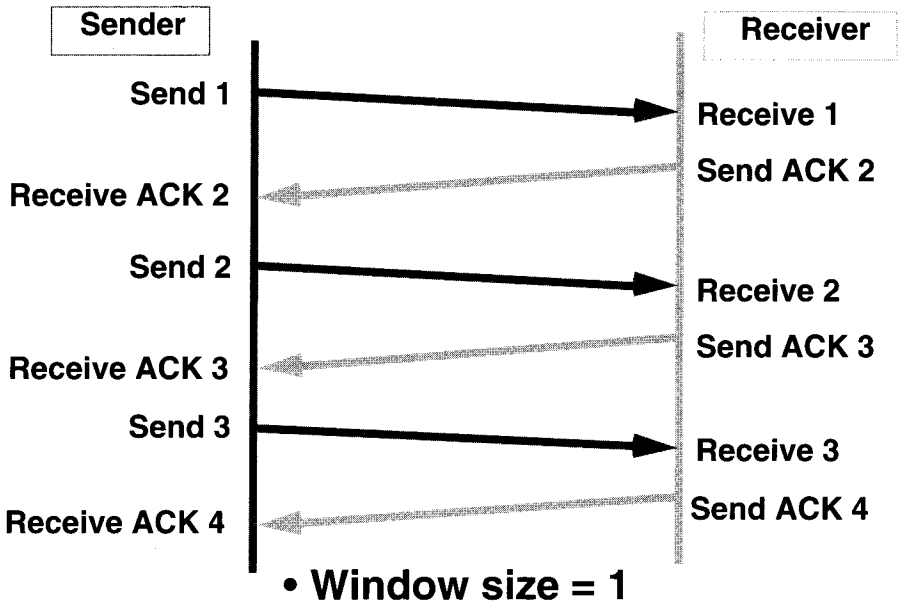


12

Both ends of the connection are synchronized with a three-way handshake/open connection sequence.

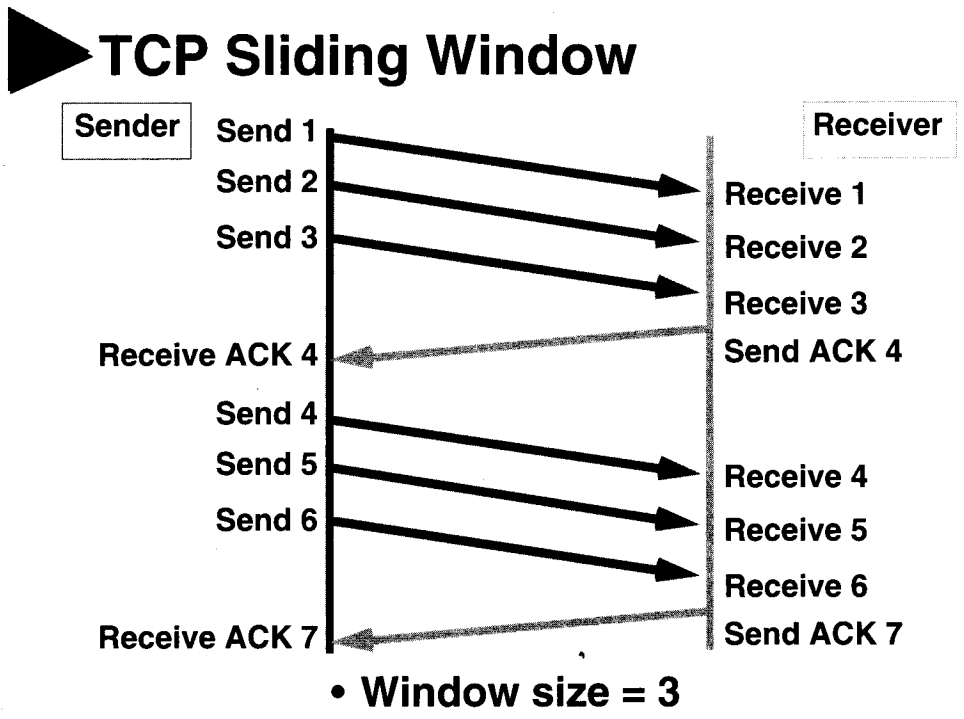
Exchanging beginning sequence numbers during the connection sequence ensures that lost data can be recovered if problems occur later.

TCP Simple Acknowledgment



13

The window size determines how much data the receiving station can accept at one time. With a window size of one, each segment must be acknowledged before another segment is transmitted. This results in inefficient use of bandwidth by the hosts.



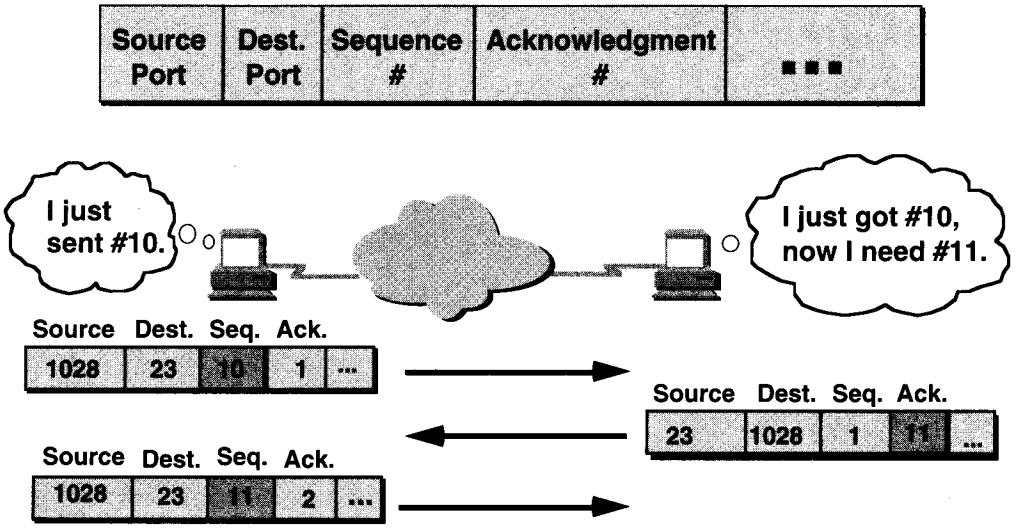
A larger window size allows more data to be transmitted pending acknowledgment.

Window size refers to the number of messages that can be transmitted while awaiting an acknowledgment. After a host transmits the window-size number of bytes, it must receive an acknowledgment before any more messages can be sent.

TCP uses expectational acknowledgments, meaning that the acknowledgment number refers to the octet expected next. The "sliding" part of "sliding window" refers to the fact that the window size is negotiated dynamically during the TCP session.

A sliding window results in more efficient use of bandwidth by the hosts.

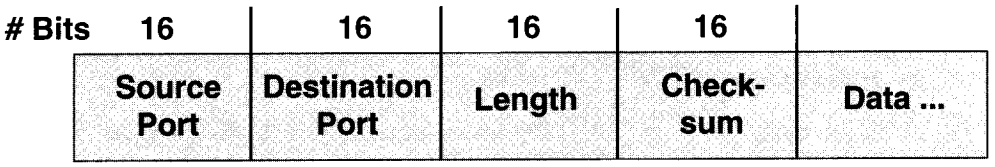
TCP Sequence and Acknowledgment Numbers



15

TCP provides sequencing of segments with a forward reference acknowledgment. Each datagram is numbered before transmission. At the receiving station, TCP reassembles the segments into a complete message. If a sequence number is missing in the series, that segment is retransmitted. Segments that are not acknowledged within a given time period result in retransmission.

UDP Segment Format

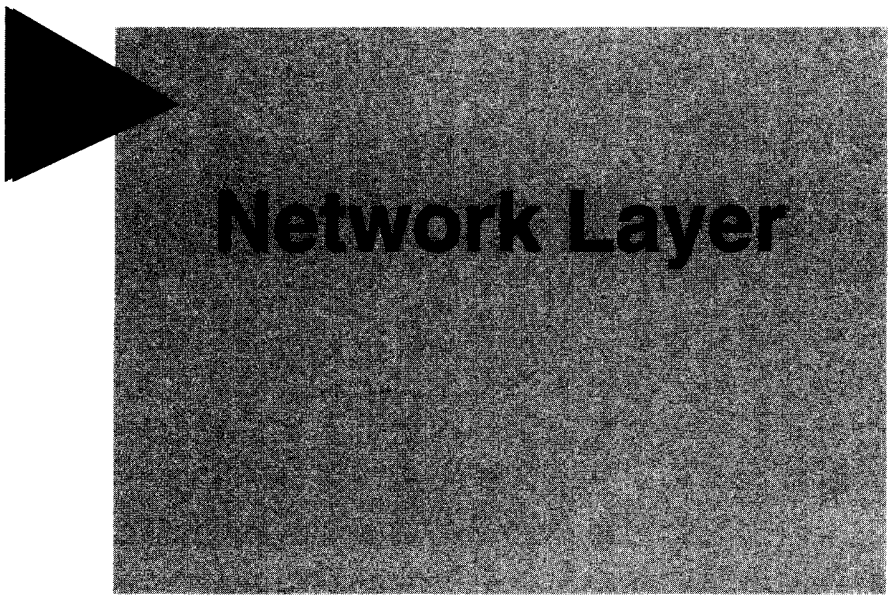


- No sequence or acknowledgment fields

16

UDP uses no windowing or acknowledgments. Application-layer protocols can provide for reliability. UDP is designed for applications that do not need to put sequences of segments together.

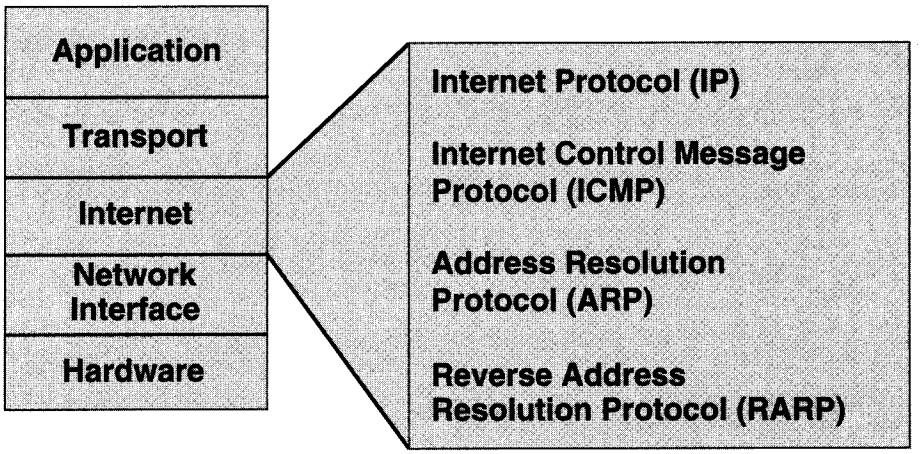
Protocols that use UDP include TFTP, SNMP, Network File System (NFS), and Domain Name System (DNS).



17

Network Layer

Network Layer Overview



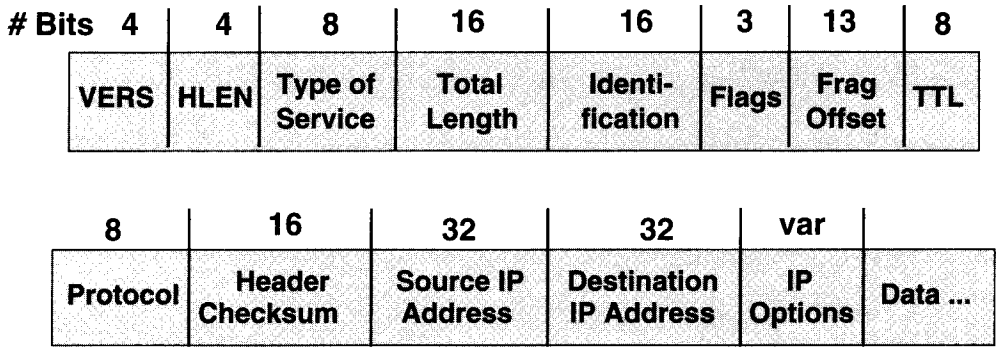
- **OSI network layer corresponds to the TCP/IP internet layer**

18

Several protocols operate at the TCP/IP internet layer, which corresponds to the OSI network layer:

- IP provides connectionless, best-effort delivery routing of datagrams. It is not concerned with the content of the datagrams. Instead, it looks for a way to move the datagrams to their destination.
- ICMP provides control and messaging capabilities.
- ARP determines the data link layer address for known IP addresses.
- RARP determines network addresses when data link layer addresses are known.

IP Datagram

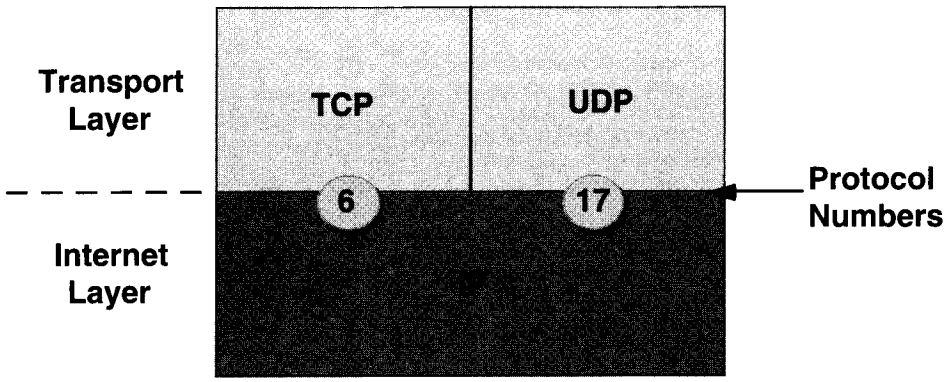


19

Field definitions within this IP datagram are as follows:

- VERS—Version number
- HLEN—Header length in 32-bit words
- Type of Service—How the datagram should be handled
- Total Length—Total length (header + data)
- Identification, Flags, Frag Offset—Provide fragmentation of datagrams to allow differing MTUs in the internet
- TTL—Time-To-Live
- Protocol—Upper-layer (Layer 4) protocol sending the datagram
- Header Checksum—Integrity check on the header
- Source and Destination IP addresses—32-bit IP addresses
- IP Options—Network testing, debugging, security, and others

Protocol Field

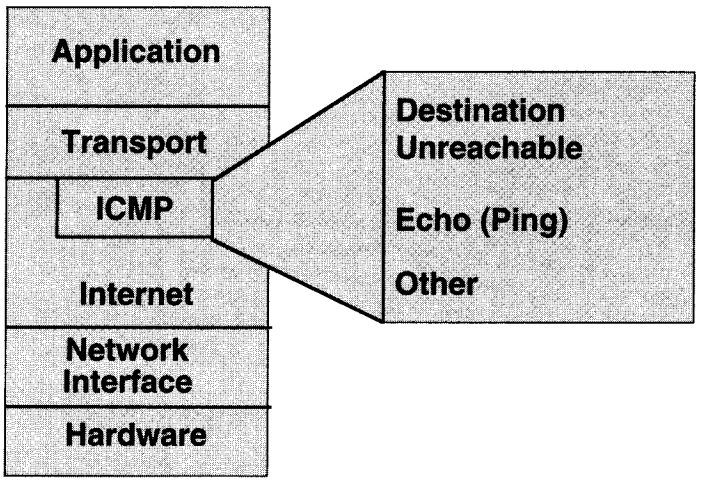


- **Determines destination upper-layer protocol**

20

The protocol field determines the Layer 4 protocol being carried within an IP datagram. Although most IP traffic uses TCP, there are other protocols that can use IP. Each IP header must identify the destination Layer 4 protocol for the datagram. Transport-layer protocols are numbered, similar to port numbers. IP includes the protocol number in the protocol field.

Internet Control Message Protocol (ICMP)

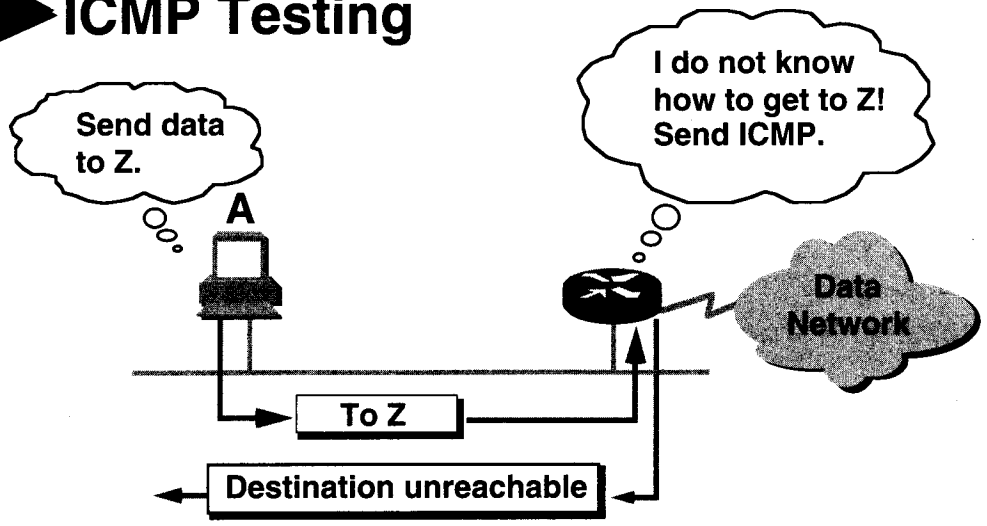


The ICMP is implemented by all TCP/IP hosts. ICMP messages are carried in IP datagrams and are used to send error and control messages.

ICMP uses the following types of defined messages. Others exist that are not included on this list:

- Destination Unreachable
- Time Exceeded
- Parameter Problem
- Source Quench
- Redirect
- Echo
- Echo Reply
- Timestamp
- Timestamp Reply
- Information Request
- Information Reply
- Address Request
- Address Reply

ICMP Testing

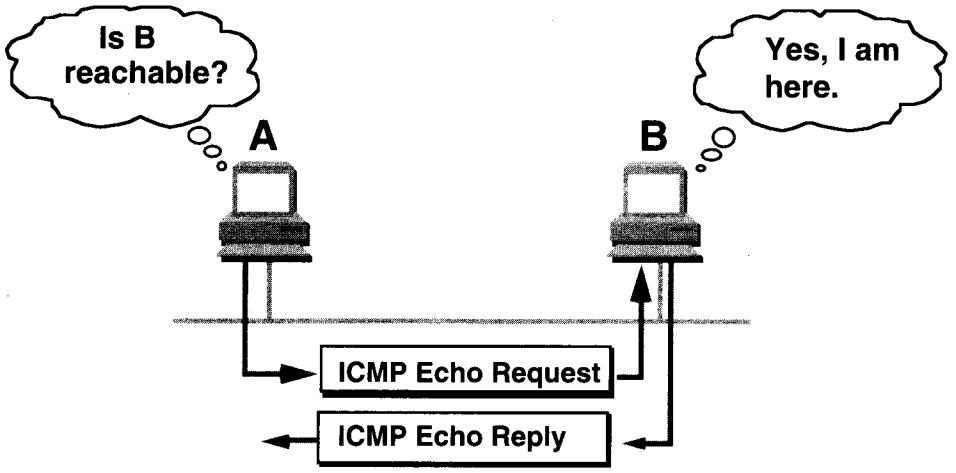


- **Destination unreachable**
  - **— Host or port unreachable**
  - **— Network unreachable**

22

If a router receives a packet that it is unable to deliver to its ultimate destination, the router sends an ICMP host unreachable message to the source. The message might be undeliverable because there is no known route to the destination.

ICMP Testing (cont.)

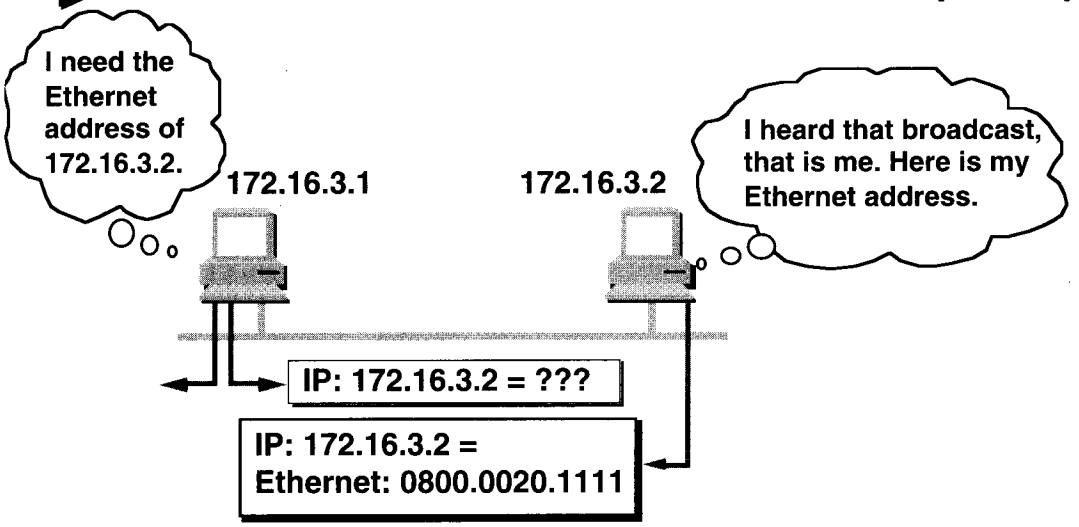


- Generated by the *ping* command

23

An echo reply is a successful reply to a **ping** command; however, results could include other ICMP messages, such as unreachables and timeouts.

Address Resolution Protocol (ARP)



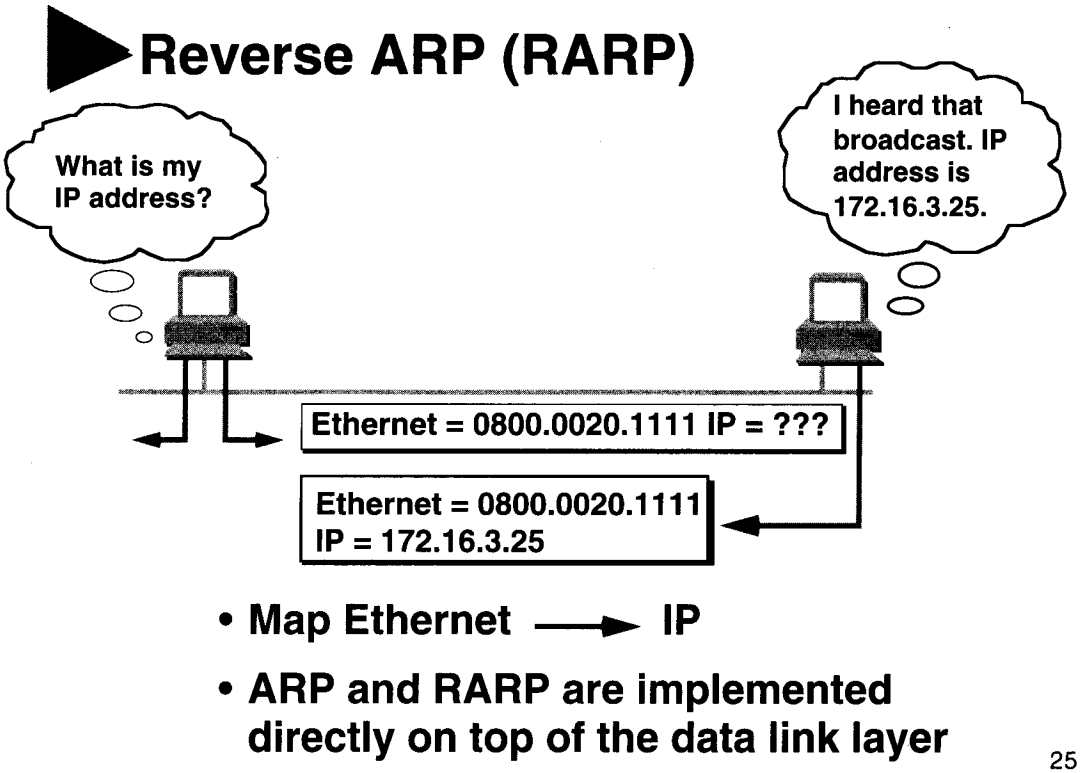
- Map IP ⟶ Ethernet
- Local ARP

24

ARP is used to resolve or map a known IP address to a MAC sublayer address to allow communication on a multiaccess medium such as Ethernet. To determine a destination address for a datagram, the ARP cache table is checked. If the address is not in the table, ARP sends a broadcast looking for the destination station. Every station on the network receives the broadcast.

The term local ARP is used to describe resolving an address when both the requesting host and the destination host share the same media or wire.

Prior to issuing the ARP, the subnet mask was consulted. The mask determined that the nodes are on the same subnet.



RARP relies on the presence of a RARP server with a table entry or other means to respond to these requests.

On the local segment, RARP can be used to initiate a remote operating system load sequence.

Summary

The TCP/IP protocol stack has the following components:

- Protocols to support file transfer, e-mail, remote login, and other applications
- Reliable and "unreliable" transports
- Connectionless datagram delivery at the network layer

ICMP provides control and message functions at the network layer

26

Exercise: TCP/IP Overview Review

Problem 1

Objective: Describe how the TCP/IP implementation relates to the OSI reference model.

Write the answer to each question.

1. Which TCP/IP reference model conceptual layer is closely related to the OSI model transport layer? _____
2. Which TCP/IP reference model conceptual layer performs functions similar to Layers 5, 6, and 7 of the OSI reference model? _____
3. Which TCP/IP reference model conceptual layer is closely related to the OSI model network layer? _____
4. Which TCP/IP reference model conceptual layer performs functions similar to Layers 1 and 2 of the OSI reference model? _____

Problem 2

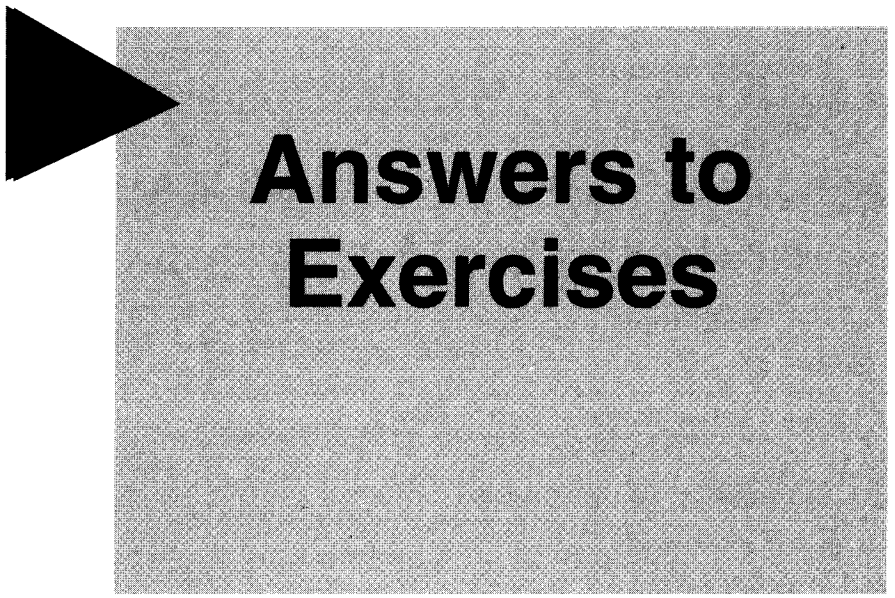
Objective: Identify the functions of the TCP/IP transport-layer protocols.

Objective: Identify the functions of the TCP/IP network-layer protocols.

Objective: Identify the functions performed by ICMP.

For each statement in the table below, write the name of the protocol being described. Place a mark in the T column if the protocol is a transport-layer protocol. Place a mark in the N column if the protocol is a network-layer protocol.

Protocol Name	T	N	Statement
			Maps a known IP address to a MAC sublayer address
			Includes Layer 4 protocol ID in header
			Used to send Destination Unreachable messages
			Breaks messages into datagrams
			Provides no software checking
			Uses sequence numbers
			Relies on application-layer reliability
			Uses table entry to respond to address requests
			Provides best-effort delivery
			Reassembles datagrams into messages
			Handshakes with receiving device
			Used to send error and control messages
			Consults subnet mask to determine whether nodes are on the same subnet
			Provides connectionless transmission
			Sends acknowledgments
			Uses no windowing



29

Answers to Exercises

Exercise: TCP/IP Overview Review

Problem 1

1. Transport layer
2. Application layer
3. Internet layer
4. Network interface layer

Problem 2

Protocol Name	T	N	Statement
ARP		x	Maps a known IP address to a MAC sublayer address
IP		x	Includes Layer 4 protocol ID in header
ICMP		x	Used to send Destination Unreachable messages
TCP	x		Breaks messages into datagrams
UDP	x		Provides no software checking
TCP	x		Uses sequence numbers
UDP	x		Relies on application layer reliability
RARP		x	Uses table entry to respond to address requests
IP		x	Provides best-effort delivery
TCP	x		Reassembles datagrams into messages
TCP	x		Handshakes with receiving device
ICMP		x	Used to send error and control messages
ARP		x	Consults subnet mask to determine whether nodes are on the same subnet
UDP	x		Provides connectionless transmission
TCP	x		Sends acknowledgments
UDP	x		Uses no windowing

CHAPTER 11

IP Address Configuration

Objectives

Upon completion of this chapter, you will be able to perform the following tasks:

Describe the different classes of IP addresses

Configure IP addresses

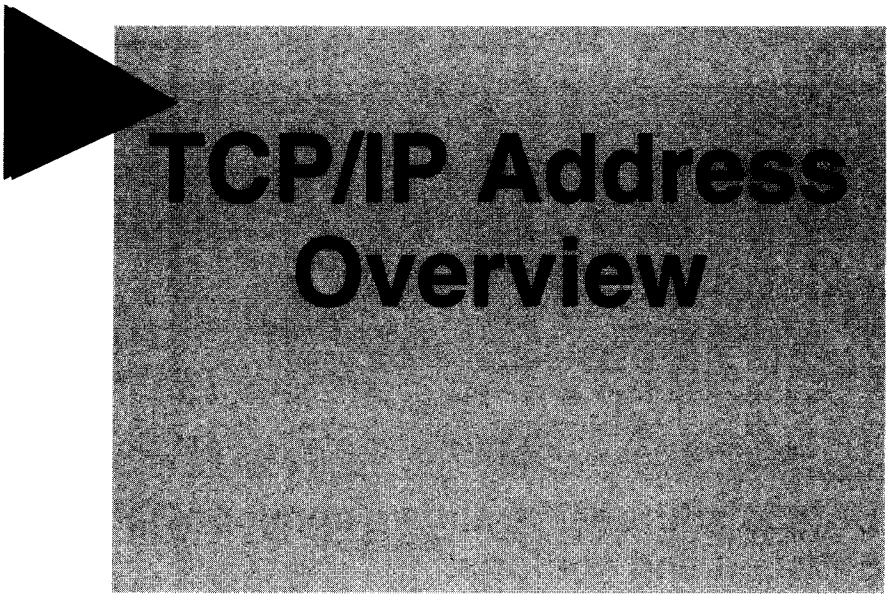
Verify IP addresses

2

This chapter discusses how IP addresses are configured. It presents address classes, subnets, subnet masks, and IP address configuration.

Sections:

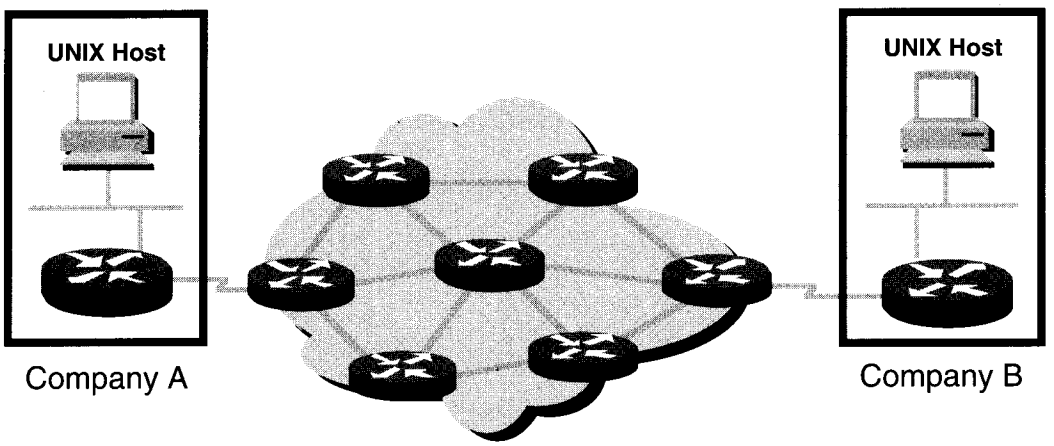
- TCP/IP Address Overview
- Configuring IP Addresses
- Answers to Exercises



3

TCP/IP Address Overview

Introduction to TCP/IP Addresses



- **Unique addressing allows communication between end stations**
- **Path choice is based on location**
- **Location is represented by an address**

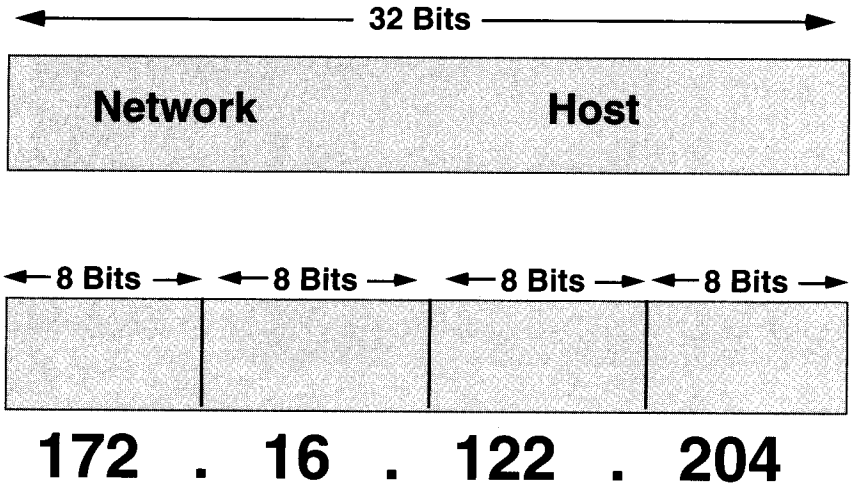
4

In a TCP/IP environment, end stations communicate seamlessly with servers or other end stations. This occurs because each node using the TCP/IP protocol suite has a unique 32-bit logical address.

Often traffic is forwarded through the internetwork based on the name of an organization, rather than an individual person or host. If names are used instead of addresses, the names must be translated to the numeric address before the traffic can be delivered. Location of the organization will dictate the path that the data follows through the internetwork.

Each company listed on the internetwork is seen as a single network that must be reached before an individual host within that company can be contacted. Each company network has an address; the hosts that populate that network share those same bits, but each host is identified by the uniqueness of the remaining bits.

IP Addressing



5

The IP address is 32 bits in length and has two parts:

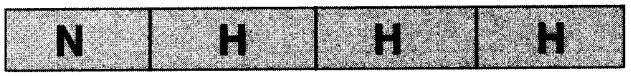
- Network number
- Host number

The address format is known as dotted decimal notation.

- Example address: 172.16.122.204.
- Each bit in the octet has a binary weight, such as (128,...4, 2, 1).
- The minimum value for an octet is 0; it contains all 0s.
- The maximum value for an octet is 255; it contains all 1s.

The allocation of addresses is managed by a central authority.

IP Address Classes

- **Class A:** 

- **Class B:** 

- **Class C:** 

- **Class D: for multicast**
- **Class E: for research**

N = Network number assigned by NIC

H = Host number assigned by network administrator 6

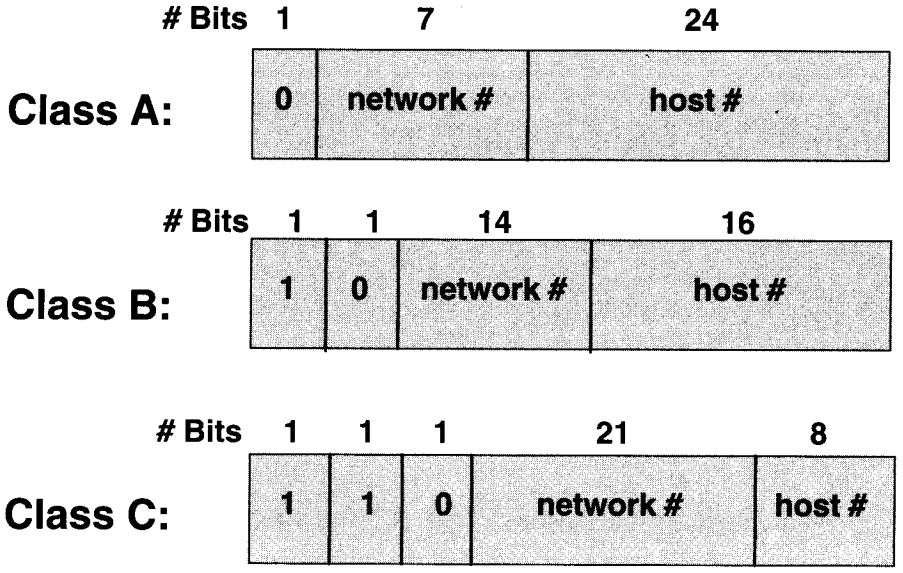
When IP was first developed, there were no classes of addresses. Now, for ease of administration, the IP addresses are broken up into classes.

There are only 126 Class A address spaces, but each one can contain approximately 16 million hosts. There are 65,534 Class B address spaces with 65,534 hosts each. There are more than 16 million Class C address spaces possible, but they only have 254 hosts each.

This scheme allows the administrative authority to assign addresses based on the size of the network. That authority designed this system on the assumption that there would be many more small networks than large networks in the world.

Note Class D and E addresses are also defined. Class D addresses start at 224.0.0.0 and are used for multicast purposes. Class E addresses start at 240.0.0.0 and are used for experimental purposes.

IP Address Bit Patterns



7

The most significant bit pattern determines the class of the address, as well as how many bits make up the network portion of the address.

RFC 1918

10.0.0.0

- Class A addresses include
  - Range of network numbers: 1.0.0.0 to 126.0.0.0
  - Number of host addresses: 16,777,214
- Class B addresses include
  - Range of network numbers: 128.1.0.0 to 191.254.0.0
  - Number of host addresses: 65,534
- Class C addresses include
  - Range of network numbers: 192.0.1.0 to 223.255.254.0
  - Number of host addresses: 254
- Class D addresses include
  - Range of network numbers: 224.0.0.0 to 239.255.255.254

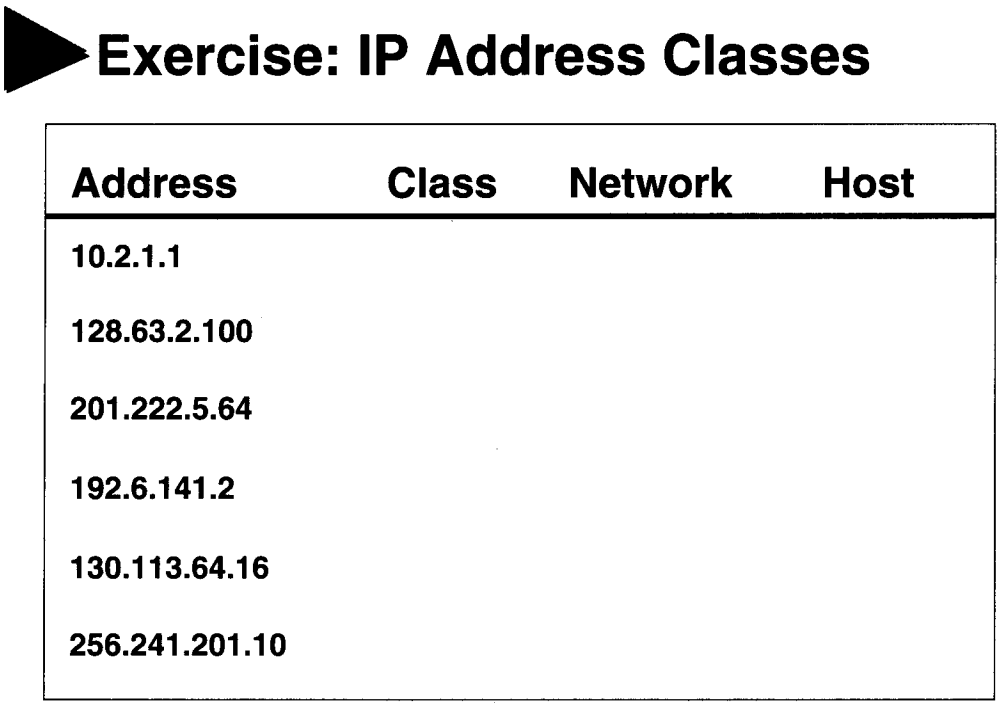
Recognizing Classes in IP Addresses (First Octet Rule)

High Order Bits	Octet in Decimal	Address Class
0	1 - 126	A
10	128 - 191	B
110	192 - 223	C

8

The first octet rule states that the class of an address can be determined by the numerical value of the first octet.

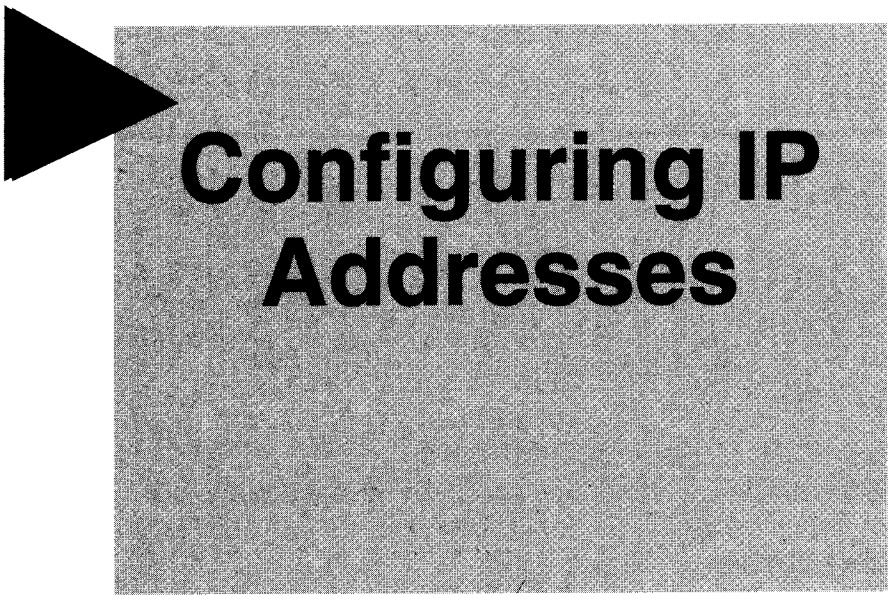
Once the first octet rule is applied, the router identifies how many bits it must match to interpret the network portion of the address (based on the standard address class). If there is no further identification of additional bits to use as part of the network address, the router can make a routing decision using this address.



Exercise: IP Address Classes

Objective: Describe the different classes of IP addresses.

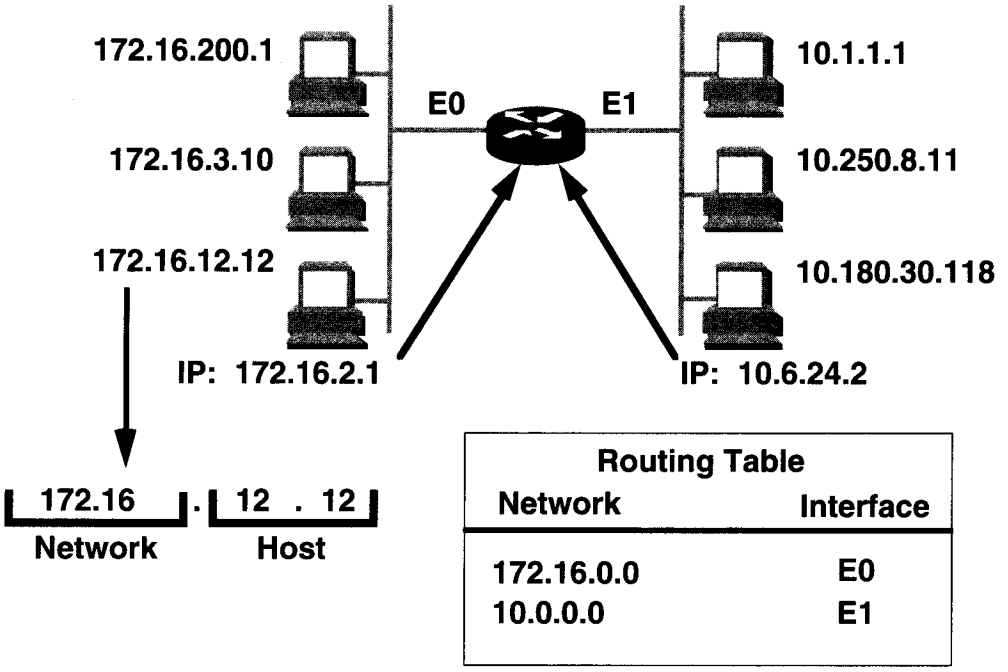
Write the address class (A, B, or C) the network, and the host numbers for each IP address listed in the table.



10

Configuring IP Addresses

Host Addresses



11

Each device or interface must have a nonzero host number.

A host address of all ones is reserved for an IP broadcast into that network.

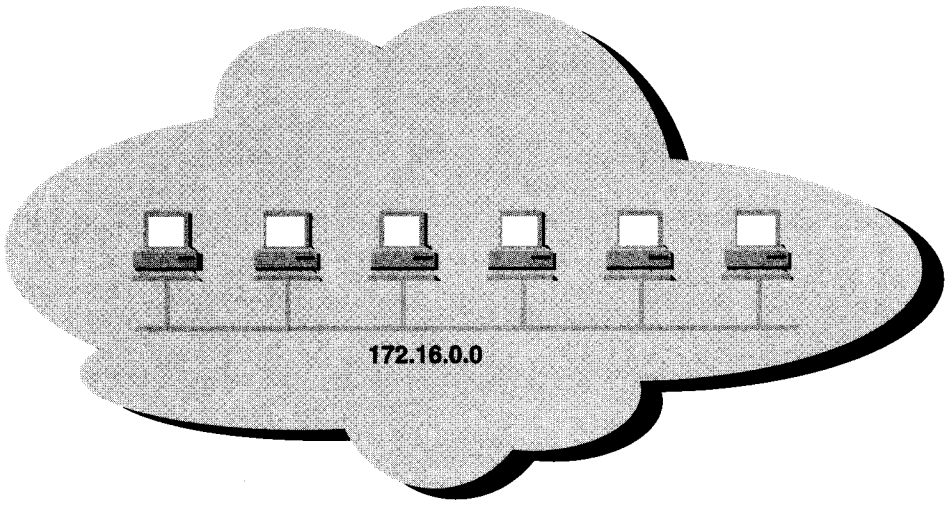
A value of zero means "this network" or "the wire itself" (for example, 172.16.0.0). It was also used for IP broadcasts in some early TCP/IP implementations, although it is rarely found now.

The routing table contains entries for network or wire addresses; it usually does not contain any information about hosts.

An IP address and subnet address on an interface achieves three purposes:

- It enables the system to process the receipt and transmission of packets.
- It specifies the device's local address.
- It specifies a range of addresses that share the cable with the device.

Addressing without Subnets



- **Network 172.16.0.0**

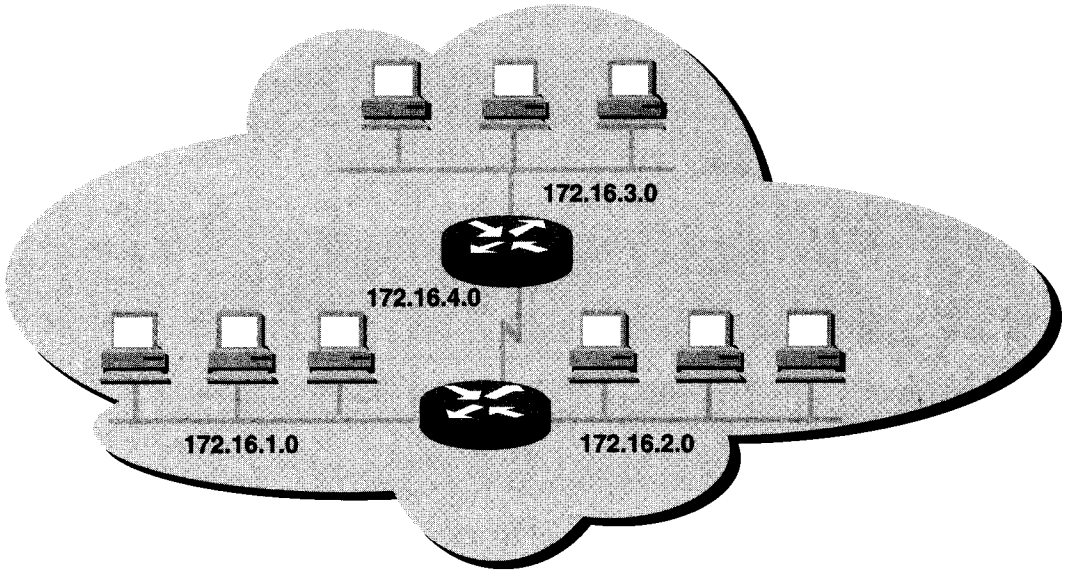
12

For an address without subnets, the outside world sees the organization as a single network, and no detailed knowledge of our internal structure is required. All datagrams addressed to 172.16 are treated the same way, regardless of the third and fourth octet of the address. A benefit from this can be the relatively short routing tables that routers can use.

Network addressing with the scheme we have set up so far has no way of distinguishing individual segments (wires) within the network. Inside the cloud having no subnets we have a single large broadcast domain—all systems on the network encounter all the broadcasts on the network. This can result in relatively poor network performance.

By default, this Class B address space defines one wire with 65,000 workstations on it. What is needed is a way to divide this wire into segments.

Addressing with Subnets



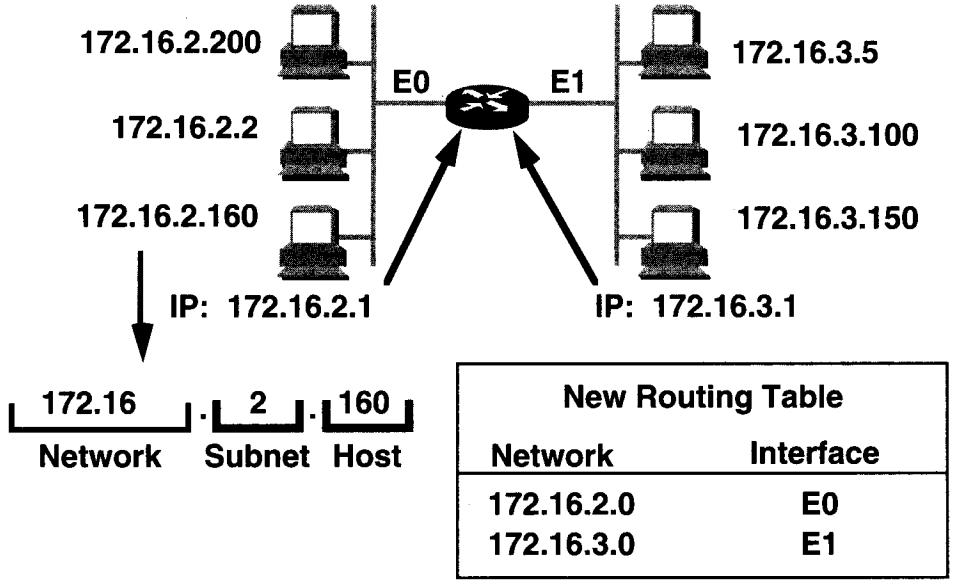
- **Network 172.16.0.0**

13

With subnets, the network address use is more efficient. There is no change to how the outside world sees the network, but within the organization, there is additional structure.

In the example, the network 172.16.0.0 is subdivided or broken up into four subnets, 172.16.1.0, 172.16.2.0, 172.16.3.0, and 172.16.4.0. Routers determine the destination network using the subnet address, limiting the amount of traffic on the other network segments.

Subnet Addressing

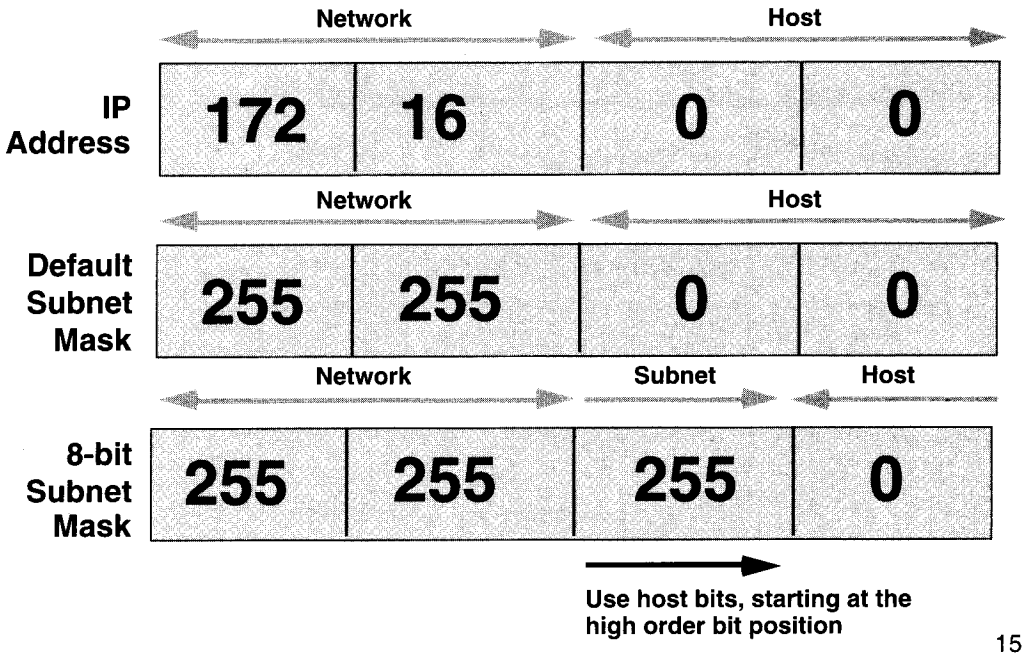


14

From the addressing standpoint, subnets are an extension of the network number. Network administrators decide the size of subnets based on organization and growth needs.

Network devices use subnet masks to identify which part of the address is considered network and which remaining part to leave for host addressing.

Subnet Mask



15

An IP address is 32 bits in size, written as four octets. The subnet mask is 32 bits in size, written as four octets. The layout of the subnet mask field is as follows:

- Binary 1 for the network bits
- Binary 1 for the subnet bits
- Binary 0 for the host bits

Subnet masks indicate which of the bits in the host field are used to specify different parts (subnets) of a particular network.

Decimal Equivalents of Bit Patterns

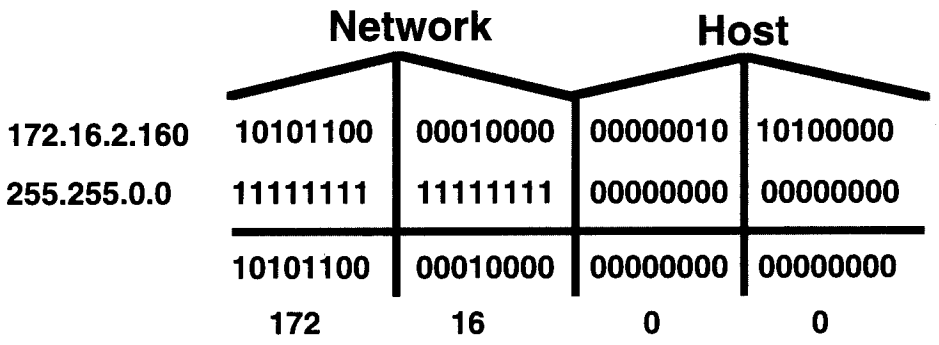
128	64	32	16	8	4	2	1		
1	0	0	0	0	0	0	0	=	128
1	1	0	0	0	0	0	0	=	192
1	1	1	0	0	0	0	0	=	224
1	1	1	1	0	0	0	0	=	240
1	1	1	1	1	0	0	0	=	248
1	1	1	1	1	1	0	0	=	252
1	1	1	1	1	1	1	0	=	254
1	1	1	1	1	1	1	1	=	255

16

Subnet bits come from the high-order bits of the host field. To determine a subnet mask for an address, add up the decimal values of each position that has a 1 in it. For example, 224 = 128 + 64 + 32.

Because the subnet mask is not defined by the octet boundary, but by bits, we need to convert dotted decimal addresses to binary and back into dotted decimal so they can work with these addresses.

Subnet Mask without Subnets



- **Subnets not in use—the default**

17

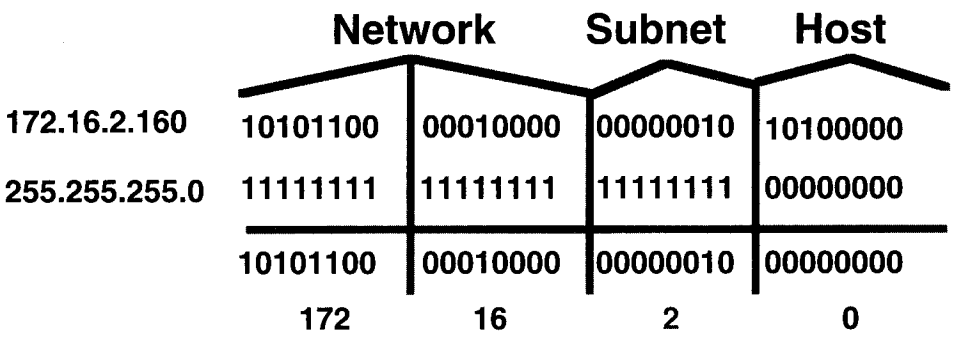
The router extracts the IP destination address from the packet and retrieves the internal subnet mask.

The router performs a logical AND operation to obtain the network number. During the logical AND operation, the host portion of the destination address is removed.

Routing decisions are then based on network number only.

In this example, with no subnetting, the network number "extracted" is 172.16.0.0.

Subnet Mask with Subnets



- **Network number extended by eight bits**

18

With eight bits of subnetting, the extracted network (subnet) number is 172.16.2.0.

This sample shows more bits turned on, extending the network portion and creating a secondary field extending from the end of the standard mask and using eight of the host bits. This secondary field is the subnet field and is used to represent wires (or subnetworks) inside the network.

Exercise: Subnet Masks

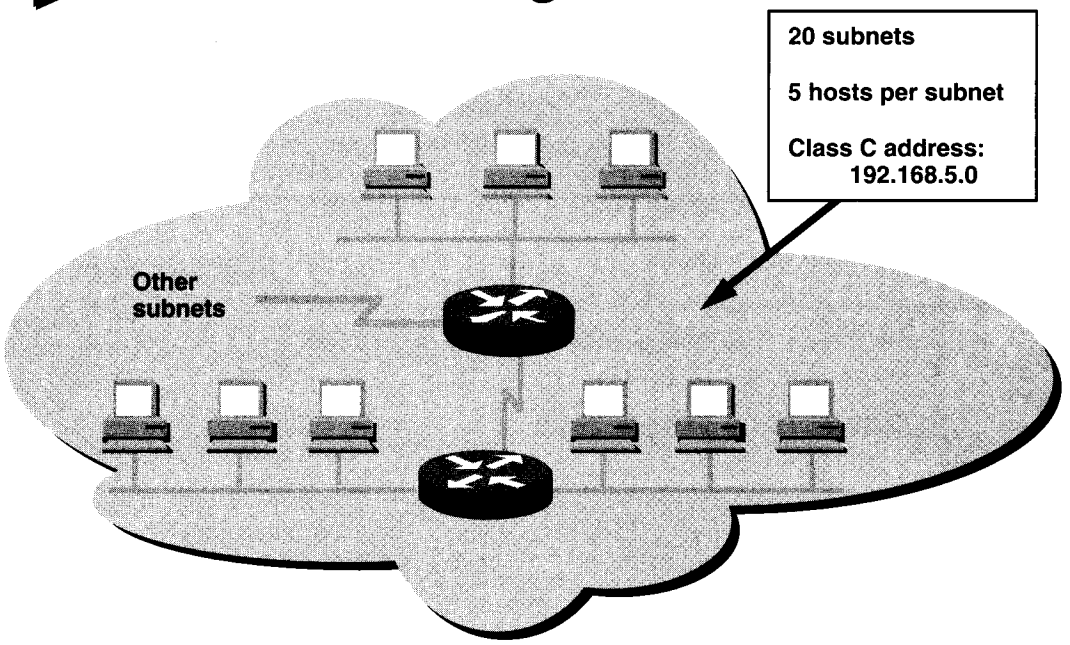
Address	Subnet Mask	Class	Subnet
172.16.2.10	255.255.255.0		
10.6.24.20	255.255.0.0		
172.30.36.12	255.255.255.0		

Exercise: Subnet Masks

Objective: Extract network information. Prepare to configure IP addresses.

Use the IP address to perform a logical AND with the subnet mask to determine the subnet number. Write the address class and subnet number in the table.

Subnet Planning



20

In this example, the network has been assigned a Class C address of 201.222.5.0. Assume 20 subnets are needed, with 5 hosts per subnet. We need to subdivide the last octet into a subnet and a host portion and determine what the subnet mask will be.

Select a subnet field size that yields enough subnetworks. In this example, choosing a 5-bit mask allows 20 subnets. In the example, the subnet addresses are all multiples of 8, such as 201.222.5.16, 201.222.5.32, and 201.222.5.48.

The remaining bits in the last octet are used for the host field. The three bits of our example allow enough hosts to cover the required five hosts per wire. The host numbers will be 1, 2, 3, and so forth.

The final host addresses are a combination of the network/subnet "wire" starting address plus each host value. The hosts on the 201.222.5.16 subnet would be addressed as 201.222.5.17, 201.222.5.18, 201.222.5.19, and so forth.

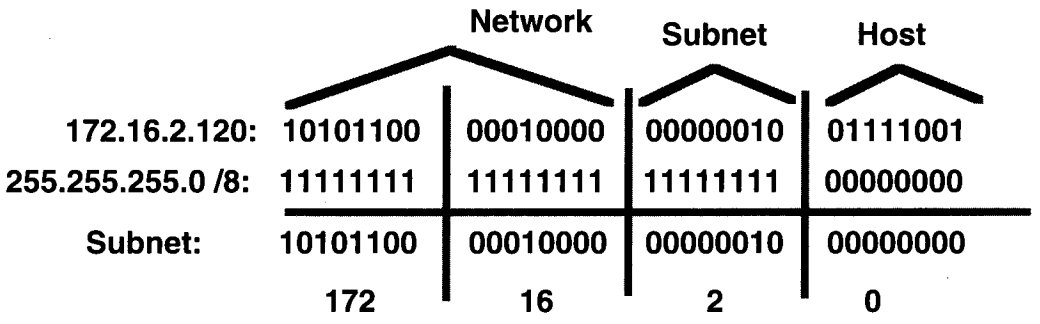
A host number of zero is reserved for the "wire" address, and a host value of all ones is reserved because it selects all hosts—a broadcast.

A table used for the subnet planning example is on the following page; also, a routing sample shows the combining of an arriving IP address with the subnet mask to derive the subnet number. The extracted subnet number should be typical of the subnets generated during this planning exercise.

Class B Subnet Planning Example

IP Host Address: 172.16.2.120

Subnet Mask: 255.255.255.0



- **Subnet Address = 172.16.2.0**
- **Host Addresses = 172.16.2.1–172.16.2.254**
- **Broadcast Address = 172.16.2.255**
- **Eight bits of subnetting** ROUTING

21

This network has eight bits of subnetting that provide up to 254 subnets and 254 host addresses.

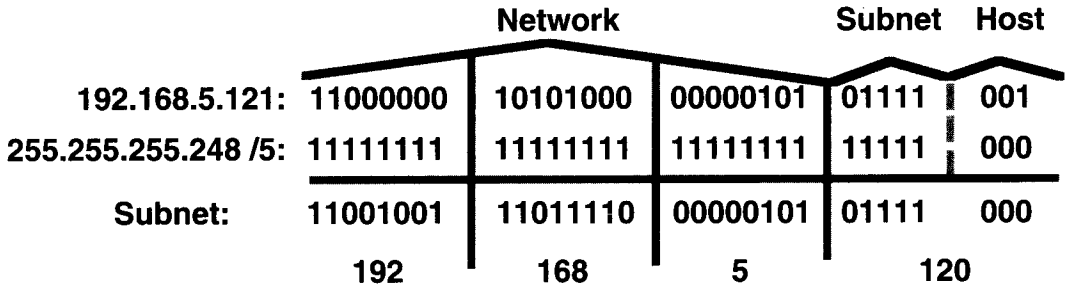
No. Bits	Subnet Mask	No. Subnets	No. Hosts
2	255.255.192.0	2	16,382
3	255.255.224.0	6	8190
4	255.255.240.0	14	4094
5	255.255.248.0	30	2046
6	255.255.252.0	62	1022
7	255.255.254.0	126	510
8	255.255.255.0	254	254
9	255.255.255.128	510	126
10	255.255.255.192	1022	62
11	255.255.255.224	2046	30
12	255.255.255.240	4094	14
13	255.255.255.248	8190	6
14	255.255.255.252	16,382	2

$2^n - 2$

Class C Subnet Planning Example

IP Host Address: 192.168.5.121

Subnet Mask: 255.255.255.248



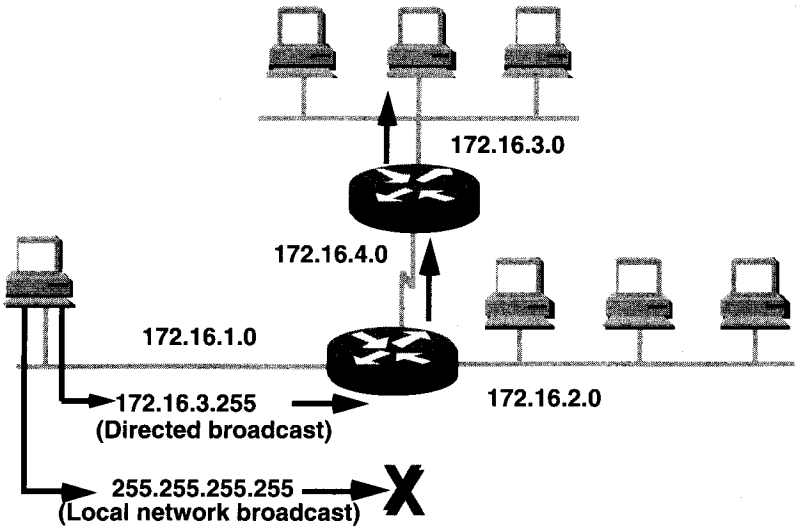
- **Subnet Address = 192.168.5.120**
- **Host Addresses = 192.168.5.121–192.168.5.126**
- **Broadcast Address = 192.168.5.127**
- **Five Bits of Subnetting**

22

In this example, a Class C network is subnetted to provide 6 host addresses and 30 subnets.

No. Bits	Subnet Mask	No. Subnets	No. Hosts
2	255.255.255.192	2	62
3	255.255.255.224	6	30
4	255.255.255.240	14	14
5	255.255.255.248	30	6
6	255.255.255.252	62	2

Broadcast Addresses



23

Broadcasting is supported on the internet. Broadcast messages are those you want every host on the network to see. The broadcast address is formed by using all ones within the IP address.

The Cisco IOS software supports two kinds of broadcasts:

- Directed broadcasts
- Flooding

Flooded broadcasts (255.255.255.255) are not propagated, but are considered local broadcasts. Broadcasts directed into a specific network are allowed and are forwarded by the router. These directed broadcasts contain all ones in the host portion of the address.

Exercise: Broadcast Addresses

Address	Subnet Mask	Class	Subnet	Broadcast
201.222.10.60/29	255.255.255.248	C	201.222.10.56	201.222.10.63
15.16.193.6/21	255.255.248.0	A		
128.16.32.13/30	255.255.255.252	B	128.16.32.12	128.16.32.15
153.50.6.27/25	255.255.255.128	B	153.50.6.0	155.50.6.127

Exercise: Broadcast Addresses

Objective: Configure IP addresses.

Write the address class, subnet number, and the broadcast address for the subnet for each of the IP addresses and subnet masks in the table.

C

201.222.10.60
255.255.255.248
201.222.10.

24 Bits of Network
5 Bits of Subnet
29 Bits of Routing

00111 | 100 = 60
11111 | 000 = 248
00111 | 000 = 56
111 = 63
Broadcast bits →
3 Bits of Host = 32

B

128.16.32.13/30
255.255.255.252
128.16.32.

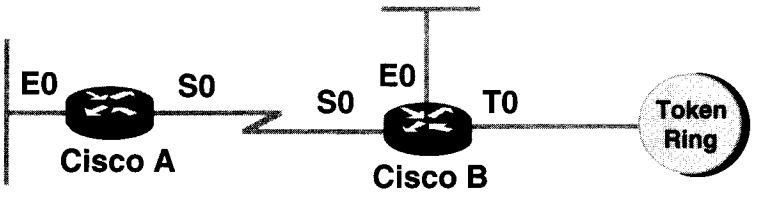
16 Bit Netwk
14 ~~16~~ Bit Sub
30 Bits of Routing

0000 1101 = 13
1111 1100 = 252
0000 1100 = 12
11 = 15
2 Bits of Host

153.50.6.27

16 Netwk
9 Sub
25 Routng

Subnetting Example



Cisco A	Mask	Subnet
E0: 172.16.2.1	255.255.255.0/8	172.16.2.0
S0: 172.16.1.1	255.255.255.0/8	172.16.1.0

Cisco B	Mask	Subnet
S0: 172.16.1.2	255.255.255.0/8	172.16.1.0
E0: 172.31.4.1	255.255.255.0/8	172.31.4.0
T0: 172.31.16.1	255.255.255.0/8	172.31.16.0

25

The graphic shows a small network with assigned interface addresses, subnet masks, and resulting subnet numbers. The number of bits in each subnet mask is indicated by the /8 following the mask.

CLASS A

8 BITS OF NETWORK

8+5 = 13 BITS OF SUBNET

21 BITS OF ROUTING

11 BITS OF HOST

TOTAL = 32

IP ADDRESS 50.50.50.50

MASK 255.255.248.0

SUBNET 50.50.48.0

BROADCAST 50.50.55.255

00110	010	= 50
11111	000	= 248
00110	000	= 48
00110	111	= 55

BROADCAST ARE 1's

IP Address Configuration

Router (config-if) #

ip address *ip-address subnet-mask*

- **Assigns an address and subnet mask**
- **Starts IP processing on an interface**

Router (config) #

term ip netmask-format [B,D,H]

- **Sets format of network mask as seen in *show* commands**

26

Use the **ip address** command to establish the logical network address of this interface.

ip address Command	Description
ip-address	A 32-bit dotted decimal number.
subnet-mask	A 32-bit dotted decimal number indicating which bit positions must match; ones indicate positions that must match, and zeros indicate positions that do not match.

Use the **term ip netmask-format** command to specify the format of network masks for the current session. Format options are:

- Bit count
- Dotted decimal (the default)
- Hexadecimal

IP Host Names

Router (config) #

ip host *name* [*tcp-port-number*] address [*address*]...

- **Defines static host name to IP address mapping**

```
ip host tokyo 1.0.0.5  2.0.0.8
ip host kyoto 1.0.0.4
```

- **Hosts/interfaces selectable by name or IP address**

27

The **ip host** command makes a static name-to-address entry in the router's configuration file.

ip host Command	**Description**
name	Any name you prefer to describe the destination.
tcp-port-number	Optional number that identifies TCP port to use when using the host name with an EXEC connect or Telnet command. The default is port 23 for Telnet.
address	IP address or addresses where the device can be reached.

In the example:

ip host tokyo 1.0.0.5 2.0.0.8	Defines two network addresses to the host **tokyo.**
ip host kyoto 1.0.0.4	Defines **kyoto** as a name equivalent for the address 1.0.0.4.

Name Server Configuration

Router (config) #

ip name-server ***server-address1*** [[***server-address2***]... ***server-address6***]

- **Specifies one or more hosts that supply host name information**

The **ip name-server** command defines which hosts can provide the name service. A maximum of six IP addresses can be specified as name servers in a single command.

To map domain names to IP addresses, you must identify the host names, then specify a name server, and enable the Domain Name System (DNS). Any time the operating system software receives a command or address it does not recognize, it refers to DNS for the IP address of that device.

Name System

Router (config) #

ip domain-lookup

- **DNS enabled by default**

Router (config) #

no ip domain-lookup

- **Turns off the name service**

IF YOU DO NOT HAVE DNS SERVER

29

Each unique IP address can have a host name associated with it. The Cisco IOS software maintains a cache of host name-to-address mappings for use by EXEC commands. This cache speeds the process of converting names to addresses.

IP defines a naming scheme that allows a device to be identified by its location in IP. A name such as ftp.cisco.com identifies the domain of the File Transfer Protocol for Cisco. To keep track of domain names, IP identifies a name server that manages the name cache.

The DNS is enabled by default with a server address of 255.255.255.255, which is a local broadcast.

The **no ip domain-lookup** command turns off name-to-address translation in the router. This means the router will not forward name system broadcast packets.

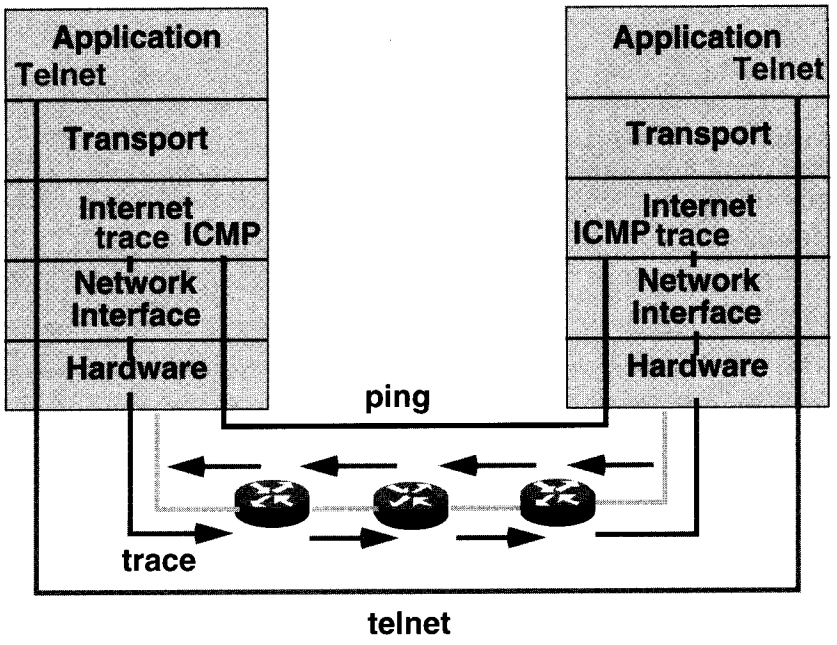
Display Host Names

```
Router# show hosts
Default domain is not set
Name/address lookup uses static mappings

Host                      Flags       Age Type   Address(es)
TOKYO                     (perm, OK)  5   IP     144.253.100.200  133.3.13.2
                                                 133.3.5.1  133.3.10.1
S                         (perm, OK) **   IP     172.16.100.155
LUBBOCK                   (perm, OK)  5   IP     183.8.128.12  153.50.3.2
AMARILLO                  (perm, OK) **   IP     153.50.129.200  153.50.3.1
BELLEVUE                  (perm, OK) **   IP     144.253.100.201  153.50.193.2
                                                 153.50.65.1  153.50.33.1
BOSTON                    (perm, OK) **   IP     144.253.100.203  192.3.63.129
                                                 192.3.63.33  192.3.63.65
CHICAGO                   (perm, OK)  5   IP     183.8.0.129  183.8.128.130
                                                 183.8.64.130
Router                    (perm, OK) **   IP     144.253.100.202  183.8.128.2
                                                 183.8.128.129  183.8.64.129
FARGO                     (perm, OK) **   IP     183.8.0.130  183.8.64.100
HARTFORD                  (perm, OK) **   IP     192.3.63.196  192.3.63.34
                                                 192.3.63.66
HOUSTON                   (perm, OK) **   IP     153.50.129.1  153.50.65.2
 --More--
```


The **show hosts** command is used to display a cached list of host names and addresses.

show hosts Command	Description
Host	Names of learned hosts.
Flags	Descriptions of how information was learned and its current status.
perm	Manually configured in a static host table.
temp	Acquired from DNS use.
OK	Entry is current.
EX	Entry has aged-out, it has expired.
Age	Time measured in hours since software referred to the entry.
Type	Protocol field.
Address(es)	Logical addresses associated with the name of the host.

Verifying Address Configuration



31

Three commands allow you to verify address configuration in your internetwork:

- **telnet**—Verifies the application-layer software between source and destination stations. This is the most complete test mechanism available.
- **ping**—Uses the ICMP protocol to verify the hardware connection and the logical address of the network layer. This is a very basic testing mechanism.
- **trace**—Uses Time-To-Live (TTL) values to generate messages from each router used along the path. This is very powerful in its ability to locate failures in the path from the source to the destination.

Simple Ping

```
Router> ping 172.16.101.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.101.1,
timeout is 2 seconds:
.!!!!
Success rate is 80 percent, round-trip min/avg/max =
6/6/6 ms
Router>
```

- **Tests IP network connectivity**

32

The **ping** command sends ICMP echo packets and is supported in both user and privileged EXEC mode. In this example, one ping timed out, as reported by the dot (.) and four were successfully received, as shown by the exclamation point (!). These are the commands that may be returned by the ping test:

Character	Definition
!	Successful receipt of an echo reply
.	Times out waiting for datagram reply
U	Destination unreachable error
C	Congestion-experienced packet
I	Ping interrupted (for example, Ctrl-Shift-6 X)
?	Packet type unknown
&	Packet Time To Live exceeded

Extended Ping

```
Router# ping
Protocol [ip]:
Target IP address: 192.168.101.162
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address:
Type of service [0]:
Set DF bit in IP header? [no]: yes
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.101.162, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/26/28 ms
Router#
```

DF = Defragment

- **Ping supported for several protocols**

33

The extended **ping** command is supported only from privileged EXEC mode.

You can use the extended command mode of the **ping** command to specify the supported internet header options. To enter the extended mode, enter Y at the extended commands prompt.

IP Trace

```
Router# trace aba.nyc.mil
Type escape sequence to abort.
Tracing the route to aba.nyc.mil (26.0.0.73)

  1  debris.cisco.com (172.16.1.6) 1000 msec 8 msec 4 msec
  2  barrnet-gw.cisco.com (172.16.16.2) 8 msec 8 msec 8 msec
  3  external-a-gateway.stanford.edu (192.42.110.225) 8 msec 4 msec 4 msec
  4  bb2.su.barrnet.net (131.119.254.6) 8 msec 8 msec 8 msec
  5  su.arc.barrnet.net (131.119.3.8) 12 msec 12 msec 8 msec
  6  moffett-fld-mb.in.mil (192.52.195.1) 216 msec 120 msec 132 msec
  7  aba.nyc.mil (26.0.0.73) 412 msec   *   664 msec
```

- **Shows interface addresses used to reach the destination**

Host names are shown if the addresses are translated dynamically or via static host table entries.

The times listed represent the time required for each of three probes to return.

Note **trace** is supported by IP, CLNS, VINES, and AppleTalk.

When the trace reaches the target destination, an asterisk (*) is reported at the display. This normally is caused by the receipt of a port-unreachable packet and the time out in response to the probe packet.

Other responses include:

!H—The probe was received by the router, but not forwarded, usually due to an access list.

P—The protocol was unreachable.

N—The network was unreachable.

U—The port was unreachable.

*—Time out.

Summary

IP addresses are specified in 32-bit dotted decimal format

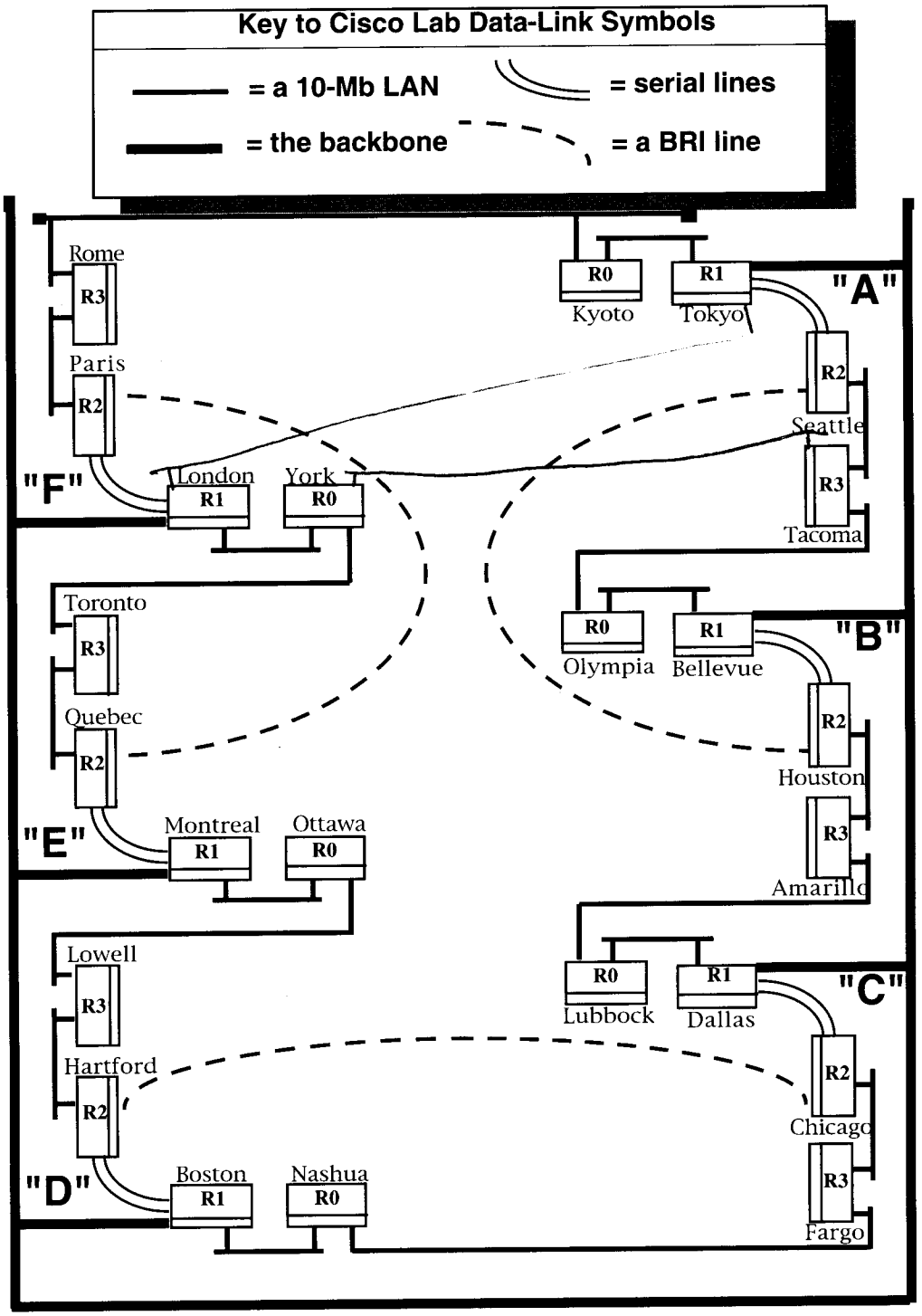
Router interfaces can be configured with an IP address

***ping* and *trace* commands can be used to verify IP address configuration**

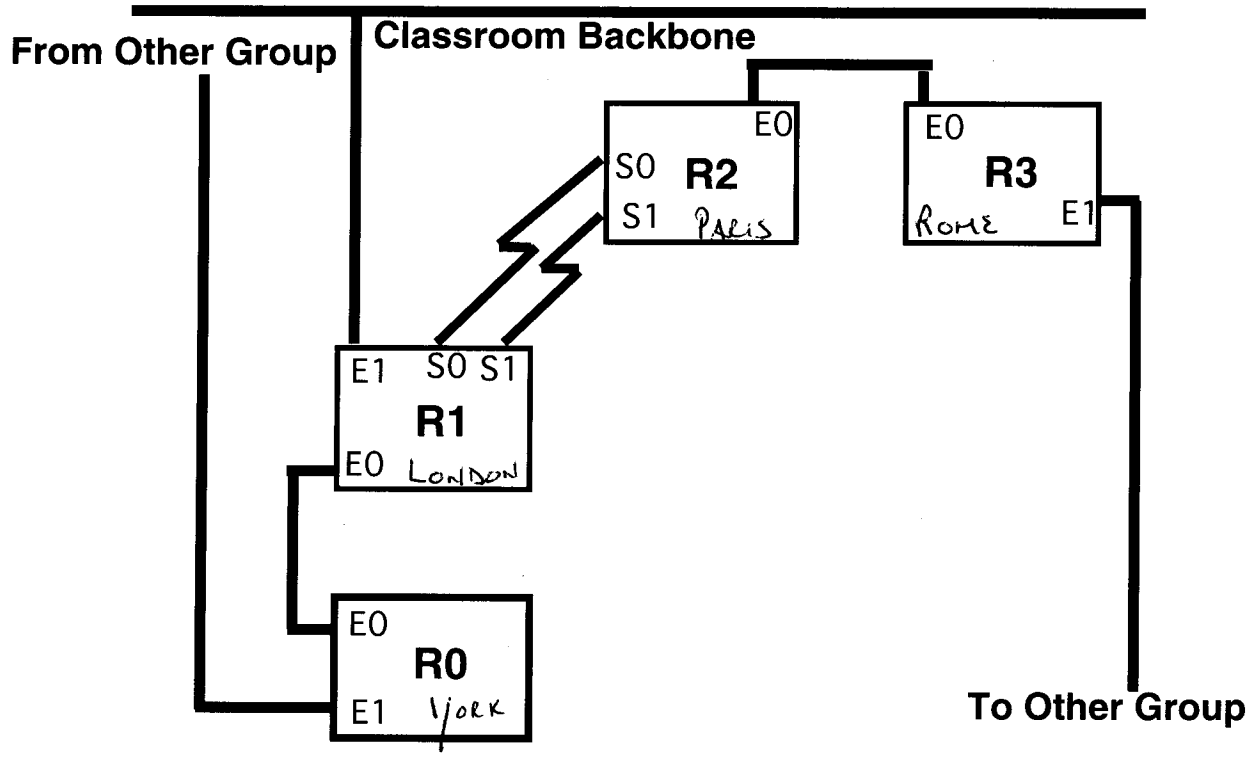
35

Lab: Network Discovery

Map of the Classroom/Lab Internetwork



Group Layout and Administrators



Objective: Describe the different classes of IP addresses.

Objective: Configure IP addresses.

Objective: Verify IP addresses.

Instructions: Introduce yourself and become acquainted with the others in your workgroup. You will be working closely with these others during the hands-on labs in this course.

Step 1 Establish a session at the console. Refer to the prompter to determine the router name. Write the router name and administrator name(s).

Group letter: F

Router name and administrator name(s) for R0:_____

Router name and administrator name(s) for R1:_____

Router name and administrator name(s) for R2:_____

Router name and administrator name(s) for R3:_____

Step 2 Discover and write the router names and administrator names for adjacent neighbors.

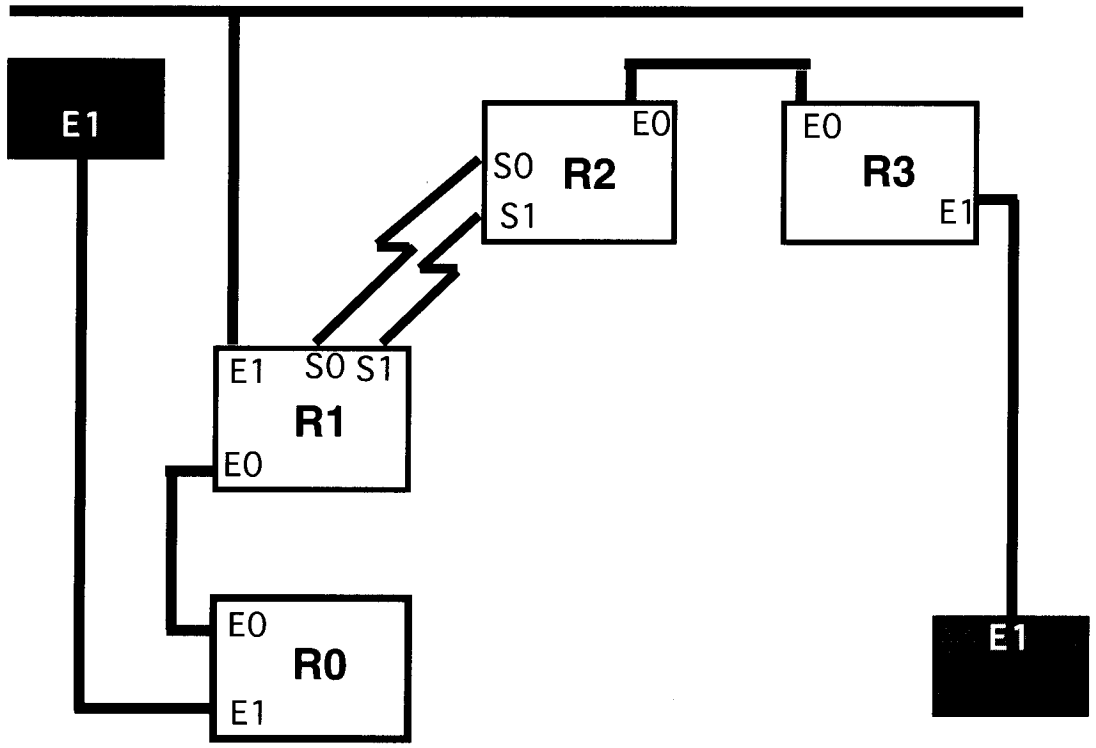
Letter of group connected to R3:_____

Router name and administrator's name for adjacent group R0:_____

Letter of group connected to R0:_____

Router name and administrator's name for adjacent group R3:_____

Network Discovery



Instructions: Use your console terminal to discover preconfigured IP addresses and subnet masks/number of subnet bits for all routers in your workgroup. Use this to calculate the subnet number for each link in your network.

Step 1 Establish a session at the console. Enter the password **cisco** if required.

Step 2 Gather information from the routers to fill in the Network Discovery worksheet with all the information you can get from the router. Note: you will calculate subnet numbers in step 5.

Step 3 To find the IP addresses of the interfaces, type **show interfaces**. Whenever you see a `---More---` prompt, press the space bar when you are ready to see the next screen. Fill in the tables on the following page with the information you discover.

Step 4 Enter the **show cdp neighbors detail** command to discover the IP addresses of your directly connected neighbors.

Step 5 Calculate subnet numbers from the information you have collected. Enter the values into the worksheet.

Step 6 Verify your completed worksheet entries with the other members of your group.

Step 7 Extra: Use **telnet** to access the other routers in your group. The password is **cisco**. Use **show** commands to see information from the perspective of the other router. To terminate this connection, type **quit**.

Group Network Discovery Worksheet

Instructions: Use the table cells in this worksheet to enter the network discovery lab information you gather from the routers in your group.

Group:

Router R0 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
E1			

Router R1 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
E1			
S0			
S1			

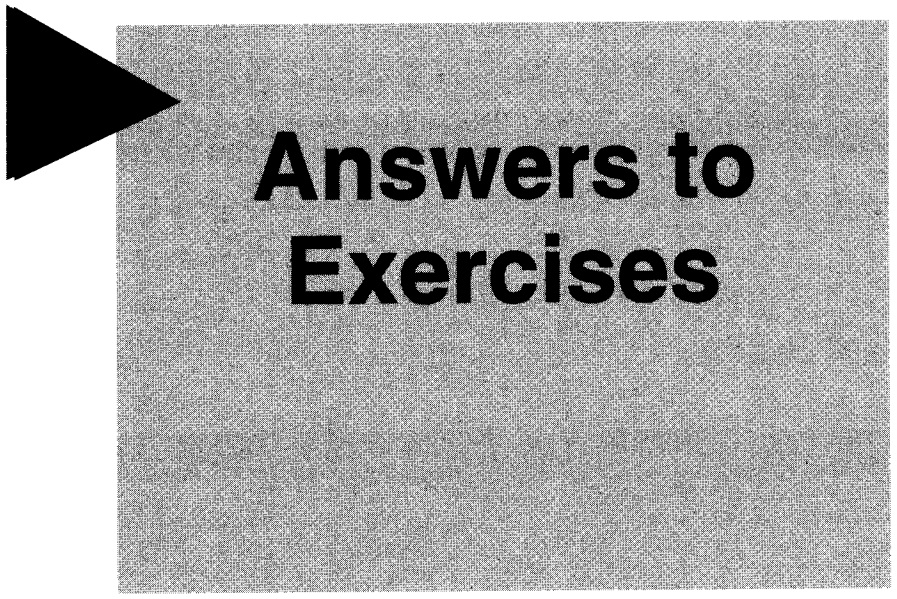
Router R2 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
S0			
S1			

Router R3 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
E1			

Group Network Discovery Notes



42

Answers to Exercises

Exercise: IP Address Classes

Address	Class	Network	Host
10.2.1.1	A	10.0.0.0	0.2.1.1
128.63.2.100	B	128.63.0.0	0.0.2.100
201.222.5.64	C	201.222.5.0	0.0.0.64
192.6.141.2	C	192.6.141.0	0.0.0.2
130.113.64.16	B	130.113.0.0	0.0.64.16
256.241.201.10	Nonexistent		

Exercise: Subnet Masks

Address	Subnet Mask	Class	Subnet
172.16.2.10	255.255.255.0	B	172.16.2.0
10.6.24.20	255.255.0.0	A	10.6.0.0
172.30.36.12	255.255.255.0	B	172.30.36.0

Bits of Routing
24
16
24

Bits of Subnet
8
16
8

Exercise: Broadcast Addresses

Address	Subnet Mask	Class	Subnet	Broadcast
201.222.10.60/29	255.255.255.248	C	201.222.10.56	201.222.10.63
15.16.193.6/21	255.255.248.0	A	15.16.192.0	15.16.199.255
128.16.32.13/30	255.255.255.252	B	128.16.32.12	128.16.32.15
153.50.6.27/25	255.255.255.128	B	153.50.6.0	153.50.6.127

CHAPTER 12

IP Routing Configuration

Objectives

Upon completion of this chapter, you will be able to perform the following tasks:

Perform the initial configuration of your router and enable IP

Add the RIP routing protocol to your configuration

Add the IGRP routing protocol to your configuration

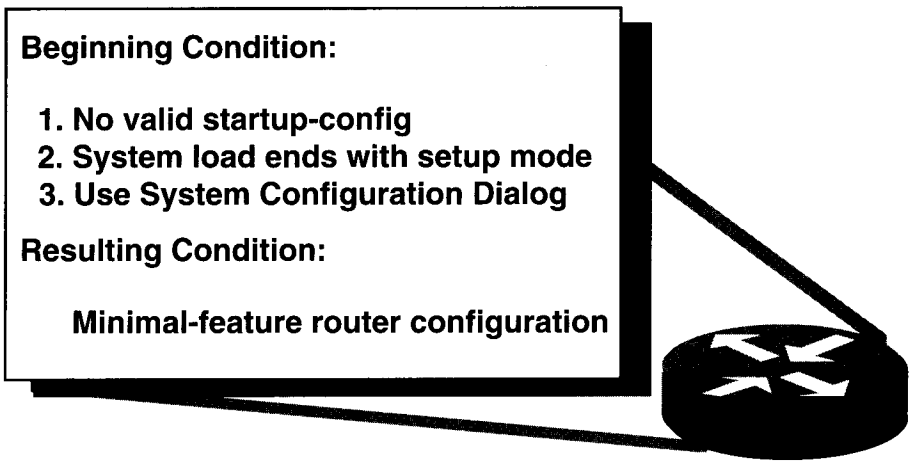
2

This chapter discusses how to configure IP routing. It includes a discussion of RIP routing and IGRP routing.

Sections:

- Configuring IP Routing
- Configuring RIP
- Configuring IGRP

Initial Router Configuration



- **Later, use *configure* to add protocol and interface changes**

3

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.

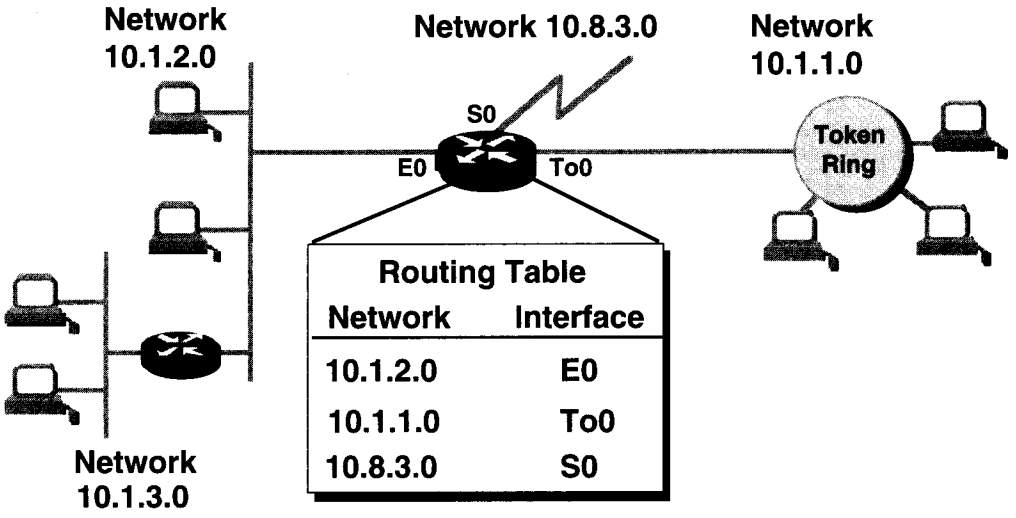
However, if the router faces a beginning condition where the router is unable to locate a valid startup-config file, it will enter an initial router configuration mode called the setup mode.

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 allows the router to use a sufficient but minimal-feature router configuration. This will include:

- An inventory of interfaces
- An opportunity to enter global parameters
- An opportunity to enter interface parameters
- A setup script review
- The 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. You can start using the router. For additional protocol and interface changes, use the enable mode and enter the command **configure**.

Initial IP Routing Table



- **Address-to-port association table**

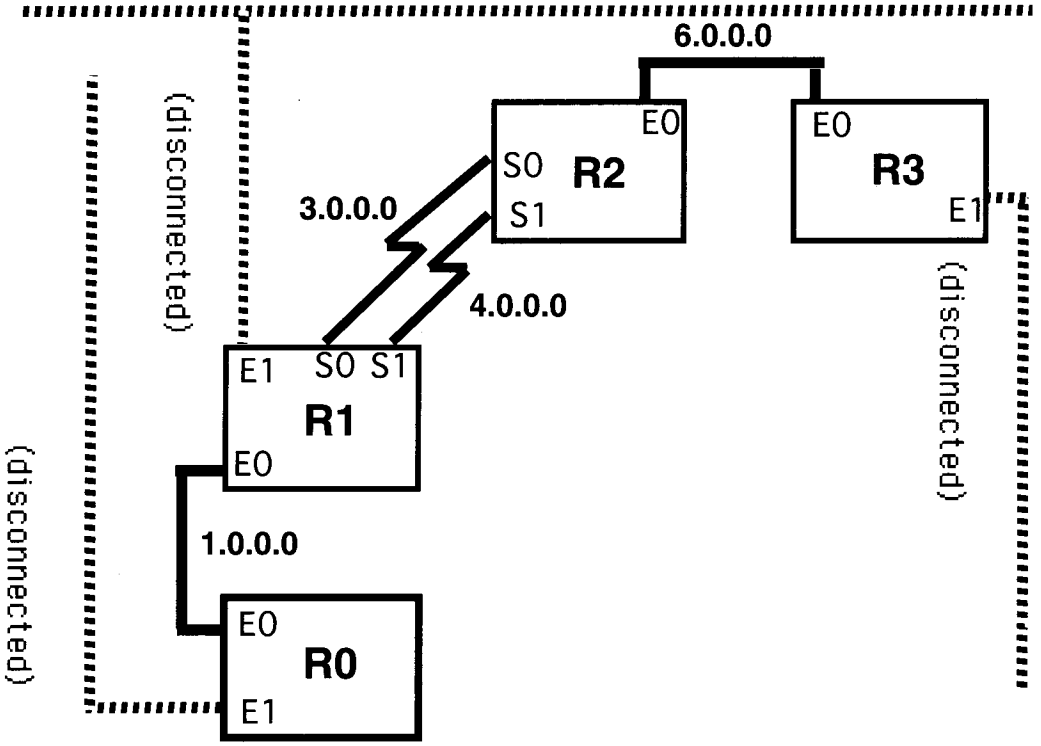
4

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

In the lab that follows, you will start up your router in a beginning condition, a state that lacks another source for 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 include address-to-port commands to set up router interfaces for IP.

Lab: Initial Router Configuration

Initial Configuration Data Sheet



Objective: Perform the initial configuration of your router and enable IP.

Instructions: Follow the procedure on the next page. You will use your console terminal to erase the startup configuration and reload all routers in your group. As each router comes up, it will enter the System Configuration Dialog—Setup. Continue the procedure to configure Cisco IOS software from scratch. Isolate your group from any other groups in class as you configure with the following addresses:

Router Role	IP Addresses for Interface
R0	E0: 1.0.0.1 Do not use any other interfaces
R1 (DCE)	E0: 1.0.0.2 S0: 3.0.0.1 S1: 4.0.0.1 Do not use any other interfaces
R2 (DTE)	E0: 6.0.0.2 S0: 3.0.0.2 S1: 4.0.02 Do not use any other interfaces
R3	E0: 6.0.0.1 Do not use any other interfaces

Use zero bits of subnetting for all interfaces (255.0.0.0).

Initial Router Configuration

Instructions: Perform the initial configuration of your router. Use the System Configuration Dialog to create an initial configuration for the router. This includes configuring the host name, password, and IP addresses of the interfaces. When done, save this configuration in nonvolatile memory.

Step 1 Initiate the System Configuration Dialog:

A) Enter the **erase startup-config** command to clear the configuration from nonvolatile RAM. (For Cisco IOS Release 10.3 and earlier, use the command **write erase.**)

B) Enter the **reload** command to initiate a reboot of the router.

The router should automatically enter setup mode. If the `router#` prompt appears, enter the **setup** command.

Step 2 Use the System Configuration Dialog to configure the following:

A) Host name to correct city name.

B) Enable password to **san-fran**.

C) Virtual terminal password to **cisco**.

D) Do not configure SNMP network management.

E) Turn on IP routing.

F) Turn off IGRP routing.

G) Turn off RIP routing.

H) Turn off all other protocols and bridging.

Step 3 Configure all the interfaces on your router as shown in the Initial Configuration Data Sheet and address table.

Remember to include any required platform-specific and serial-line specific parameters. Do not configure *IP unnumbered* on serial interfaces.

Step 4 Check the configuration script generated by the System Configuration Dialog; if it looks correct, save the configuration by entering **yes** at the prompt.

Step 5 Enter enable mode.

Step 6 Use the **configure terminal** command to set the console password to **cisco**.

Step 7 Enter the config-interface environment and enable CDP on each of your router interfaces by entering the command **cdp enable**.

Step 8 Quit the session and log in again. The router should ask for a password.

Step 9 After testing the passwords, save the configuration in nonvolatile memory.

Initial Group Connectivity Testing

Instructions: After your initial configuration on the router, use the interface configuration mode to close the parallel serial link. Then use Telnet to determine the connectivity of your router to all the other routers in your group.

Step 1 On the R1 and R2 routers, use the **configure terminal** command to **shut** the interfaces to network 3.0.0.0.

Step 2 Attempt to **telnet** to the other routers in your group.

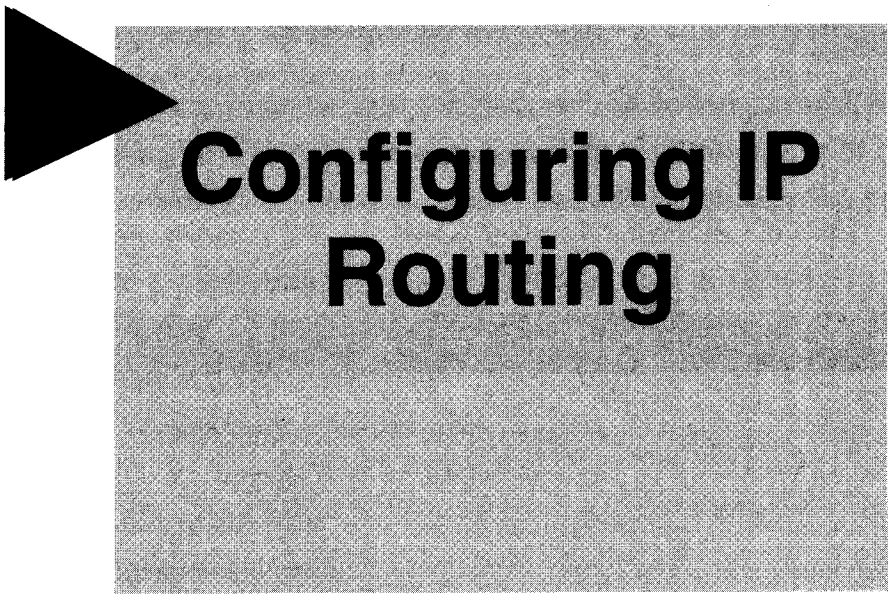
Here is the problem: Are some addresses reachable while others are unreachable?

Step 3 Record the interface addresses of the routers that you try to reach from your router and the results of your efforts in the following table.

Router Role/Name	Destination IP Addresses	Reachable?	Unreachable?
R0/			
R1/			
R2/			
R3/			

Step 4 Compare the routers you can reach with the other members of your group. *What is going on here? Can you explain the behavior of the router in this case?* Write the consensus explanation from your group for this behavior.

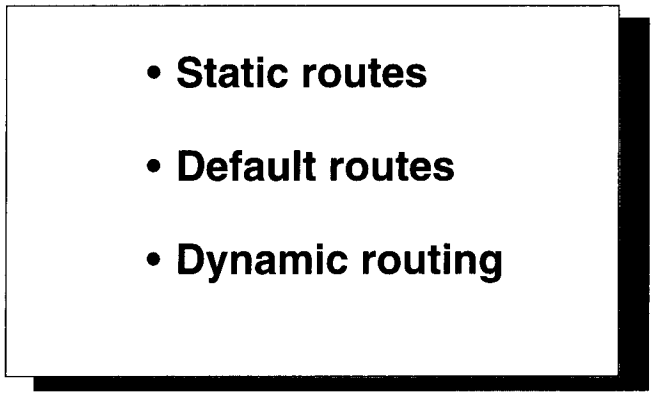
Step 5 Use the **show ip route** command to examine the routing table. *Does this help explain the behavior?* Write the consensus interpretation from your group for this command output.



8

Configuring IP Routing

IP Routing Learns Destinations



9

Default routers learn paths to destinations three different ways:

- Static routes—Manually defined by the system administrator as the only path to the destination. Useful for controlling security and reducing traffic.
- Default routes—Manually defined by the system administrator as the path to take when no route to the destination is known.
- Dynamic routing—Router learns of paths to destinations by receiving periodic updates from other routers.

Static Route Configuration

Router (config) #

ip route *network* [*mask*] { *address* | *interface* } [*distance*]

- **Defines a path to an IP destination network or subnet**

The **ip route** command sets up a static route.

ip route Command	Description
network	Destination network or subnet. NOT AN INTERFACE
mask	Subnet mask.
address	IP address of next hop router.
interface	Name of interface to use to get to destination network.
distance	The administrative distance.

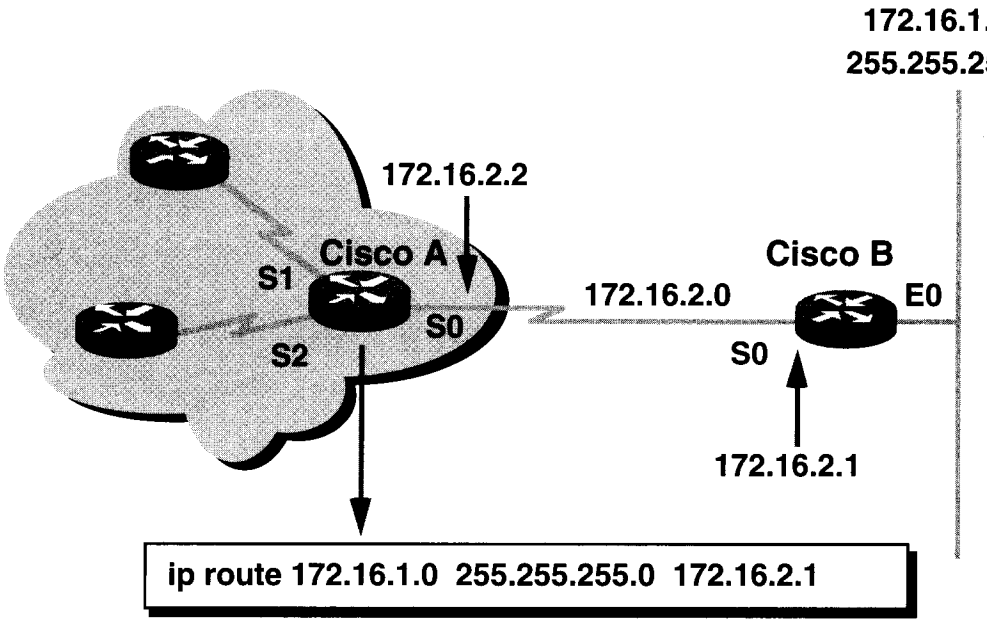
The administrative distance is a rating of the trustworthiness of a routing information source expressed as a numeric value from 0 to 255. The higher the number, the lower the trustworthiness rating.

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.

Routing updates are not sent on a link if only defined by a static route, thereby conserving bandwidth.

Static Route Example



In the example:

ip route 172.16.1.0 255.255.255.0 172.16.2.1 Command	Description
ip route 172.16.1.0	Specifies a static route to the destination subnetwork.
255.255.255.0	Subnet mask indicates that 8 bits of subnetting are in effect.
172.16.2.1	IP address of next hop router in the path to the destination.

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, so a default route may be more appropriate.

Default Route Configuration

Router (config) #

ip default-network *network-number*

- **Defines a default route**

12

The **ip default-network** command establishes a default route.

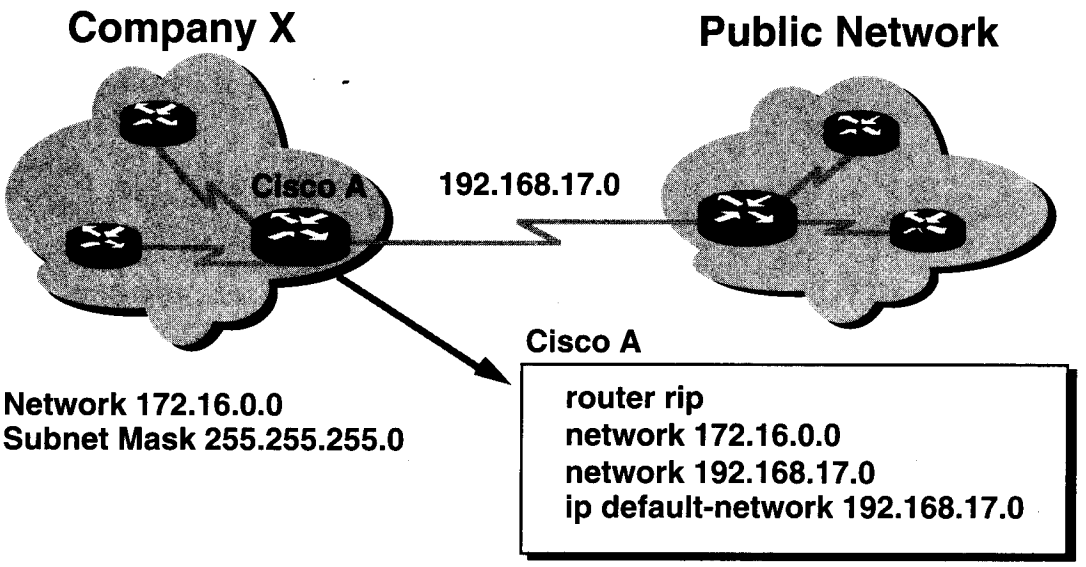
ip default-network Command	Description
network-number	IP network number or subnet number defined as the default.

When an entry for the destination network does not exist in the routing table, the packet is sent to the default network. The default network must exist in the routing table. Default routes keep the length of routing tables shorter.

Use the default network number when you need a route but have only partial information about the destination network.

Because the 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.

Default Route Example



13

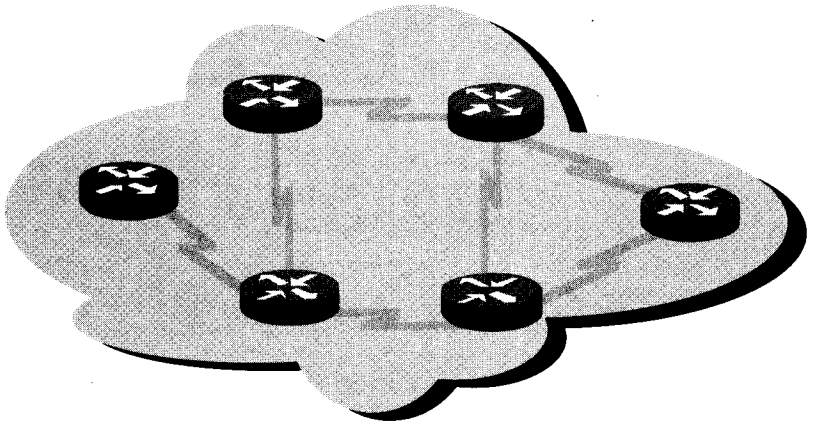
In the example, the global command **ip default-network 192.168.17.0** defines the Class B network 192.168.17.0 as the destination path for packets that have no routing table entry.

Router A could need a firewall for routing updates. The Company X administrator does not want updates coming in from the public network. 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.

ROUTER RIP

PASSIVE INTERFACE S0 . (DO NOT SEND RIP UPDATES OUT THIS INTERFACE.)

Autonomous System



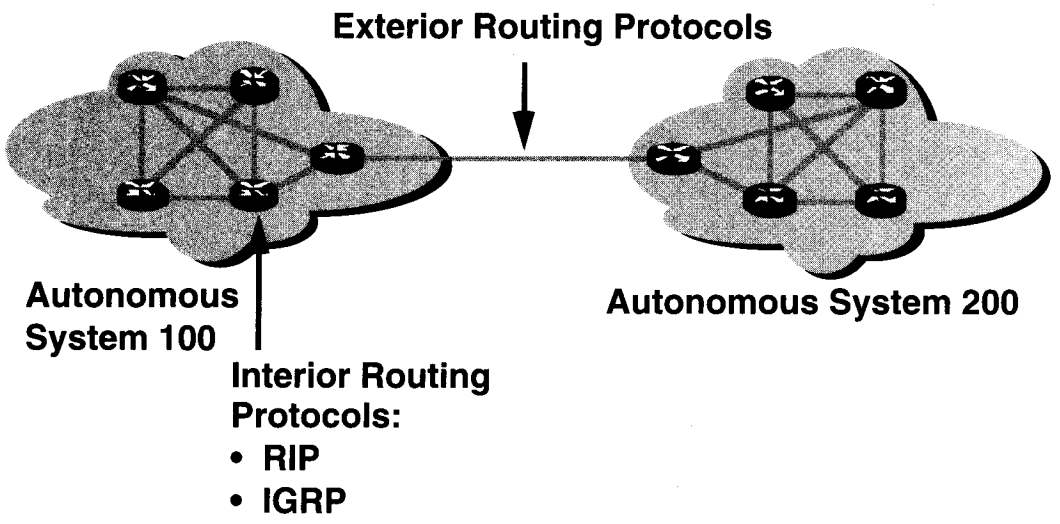
- **Routers under a common administration**

14

On the previous page, you saw how Company X used a default router to connect to a public network with a routing protocol that uses an autonomous system. An autonomous system consists of routers, run by one or more operators, that present a consistent view of routing to the external world.

The Network Information Center (NIC) assigns a unique autonomous system to enterprises. This autonomous system is a 16-bit number. A routing protocol such as Cisco's Interior Gateway Routing Protocol (IGRP) requires that you specify this unique, assigned autonomous system number in your configuration.

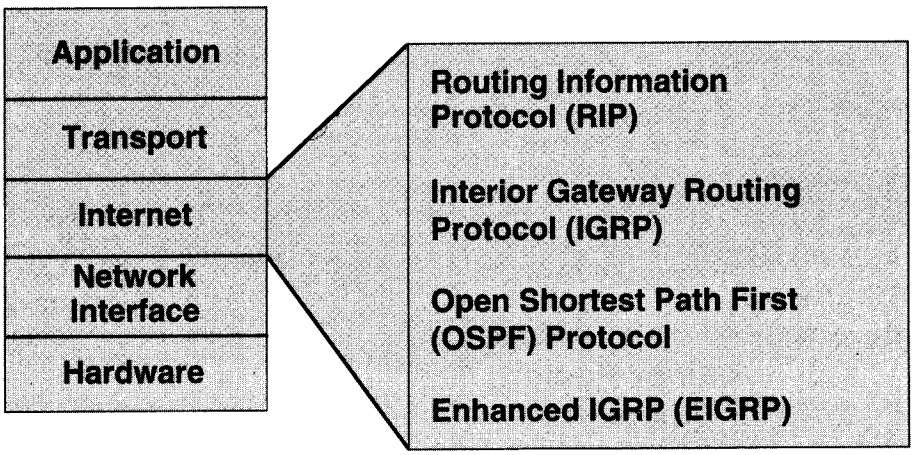
Interior or Exterior Routing Protocols



15

Exterior routing protocols are used to communicate between autonomous systems. Interior routing protocols are used within a single autonomous system.

Interior IP Routing Protocols



16

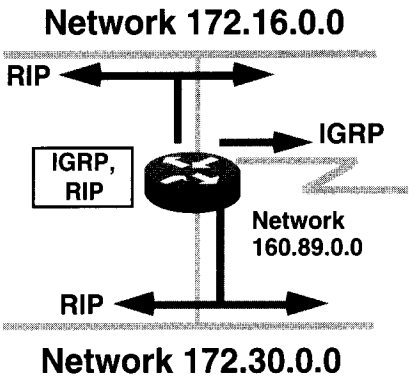
At the internet layer of the TCP/IP suite of protocols, a router can use the IP routing protocol to accomplish routing through the implementation of a specific routing algorithm. Examples of the IP routing protocols include:

- RIP—A distance vector routing protocol
- IGRP—Cisco's distance vector routing protocol
- OSPF—A link-state routing protocol
- Enhanced IGRP—A balanced hybrid routing protocol

The following pages teach you how to configure the first two of these protocols in class.

IP Routing Configuration Tasks

- **Global configuration**
  - **— Select routing protocol(s)**
  - **— Specify network(s)**
- **Interface configuration**
  - **— Verify address/subnet mask**



17

The selection of IP as a routing protocol involves the setting of both global and interface parameters.

Global tasks:

- Select a routing protocol, RIP or IGRP.
- Assign IP network numbers without specifying subnet values.

The interface task is to assign network/subnet addresses and the appropriate subnet mask.

Dynamic routing uses broadcasts and multicasts to communicate with other routers.

The routing metric helps routers find the best path to each network or subnet.

Dynamic Routing Configuration

Router (config) #

router *protocol* [*keyword*]

- **Defines an IP routing protocol**

Router (config-router) #

network *network-number*

- **The network subcommand is a mandatory configuration command for each IP routing process**

18

The **router** command starts a routing process.

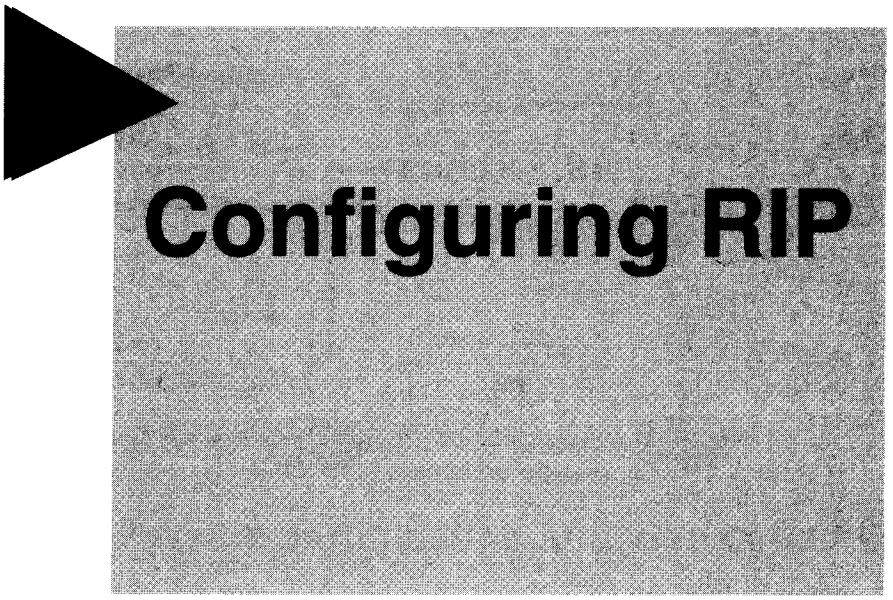
router Command	Description
protocol	Either RIP, IGRP, OSPF, or Enhanced IGRP.
keyword	Such as autonomous system, which is used with those protocols that require an autonomous system, such as IGRP.

For RIP To work:

The **network** command is required because it allows the routing process to determine which interfaces will participate in the sending and receiving of routing updates.

network Command	Description
network-number	Specifies a directly connected network.

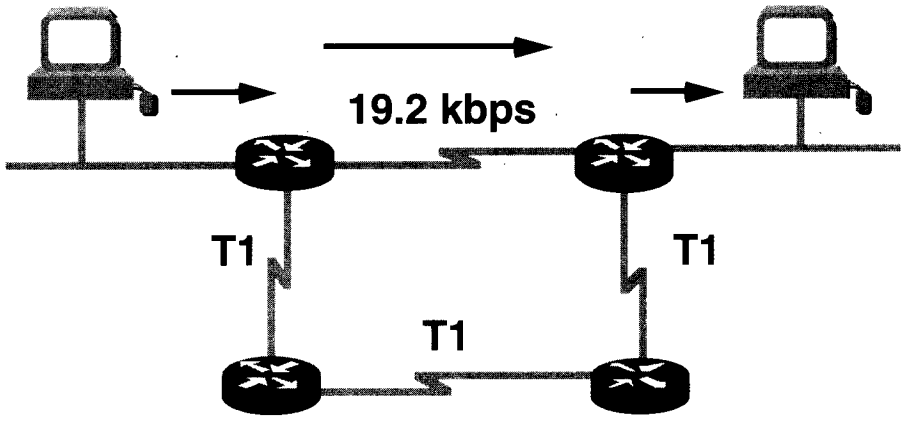
The network number must be based on the NIC network numbers, not subnet numbers or individual addresses.



19

Configuring RIP

RIP Overview



- **Hop count metric selects the path**

20

The RIP protocol was originally specified in RFC 1058.

Key characteristics of RIP include the following:

- It is a distance vector routing protocol.
- Hop count is used as the metric for path selection.
- The maximum allowable hop count is 15.
- Routing updates are broadcast every 30 seconds by default.

RIP Configuration

Router (config) #

router rip

- **Starts the RIP routing process**

Router (config-router) #

network *network-number*

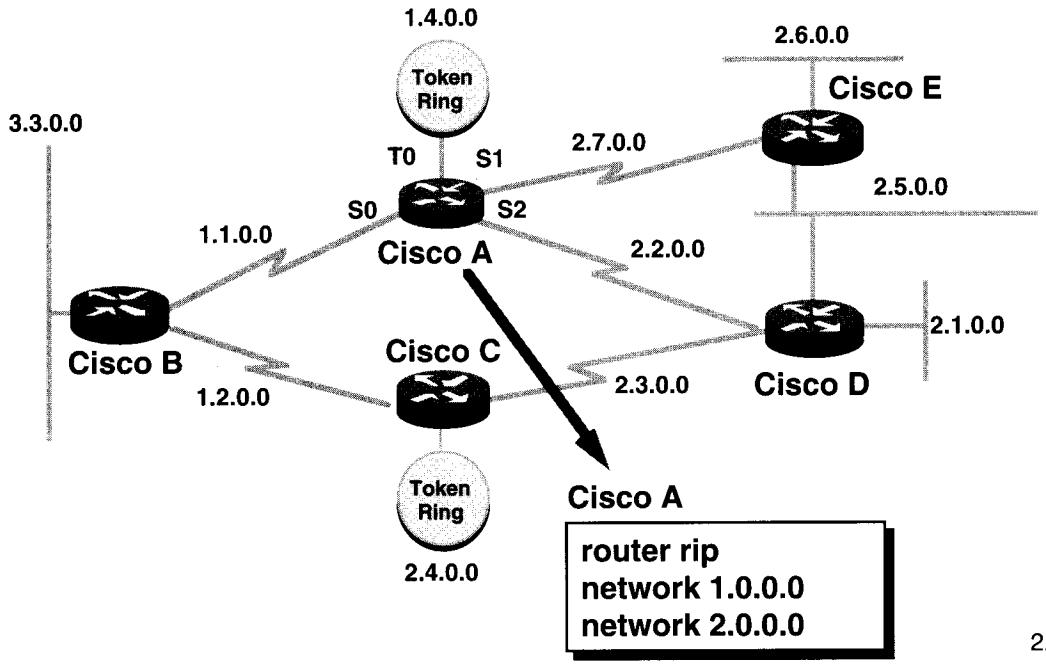
- **Selects participating attached networks**

21

The **router rip** command selects RIP as the routing protocol.

The **network** command assigns a NIC-based address to which the router is directly connected. The routing process will associate interfaces with the proper addresses and will begin packet processing on the specified networks.

RIP Configuration Example



22

In the example:

- **router rip**—Selects RIP as the routing protocol.
- **network 1.0.0.0**—Specifies a directly connected network.
- **network 2.0.0.0**—Specifies a directly connected network.

The Cisco A router interfaces connected to networks 1.0.0.0 and 2.0.0.0 will send and receive RIP updates. These routing updates allow the router to learn the network topology.

Monitoring IP

```
Router> show ip protocol
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 13 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Routing for Networks:
    183.8.0.0
    144.253.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    183.8.128.12         120      0:00:14
    183.8.64.130         120      0:00:19
    183.8.128.130        120      0:00:03
  Distance: (default is 120)
```


The **show ip protocol** command displays values about routing timers and network information associated with the entire router. Use this information to identify a router that is suspected of delivering bad routing information.

This router sends updated routing table information every 30 seconds. (This interval is configurable.) It has been 17 seconds since it sent its last update, and the next one will be sent in 13 seconds.

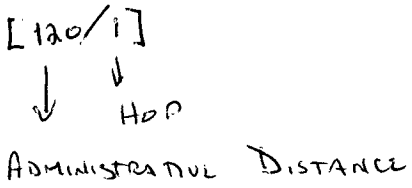
The router is injecting routes for the networks listed following the Routing for Networks line.

Displaying the IP Routing Table

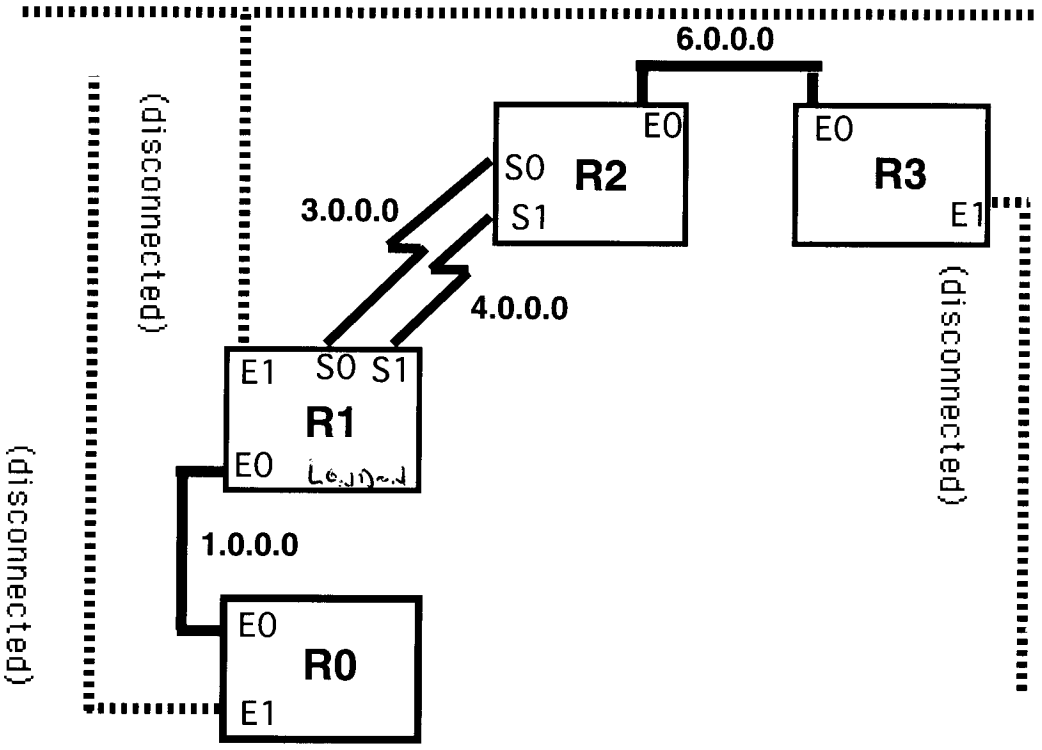
```
Router> show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default

Gateway of last resort is not set

     144.253.0.0 is subnetted (mask is 255.255.255.0), 1 subnets
C       144.253.100.0 is directly connected, Ethernet1
R    133.3.0.0
R    153.50.0.0 [120/1] via 183.8.128.12, 00:00:09, Ethernet0
     183.8.0.0 is subnetted (mask is 255.255.255.128), 4 subnets
R       183.8.0.128 [120/1] via 183.8.128.130, 00:00:17, Serial0
                    [120/1] via 183.8.64.130, 00:00:17, Serial1
C       183.8.128.0 is directly connected, Ethernet0
C       183.8.64.128 is directly connected, Serial1
C       183.8.128.128 is directly connected, Serial0
R    192.3.63.0
```


The **show ip route** command displays the contents of the IP routing table.

The routing table contains entries for all known networks and subnetworks and contains a code that indicates how that information was learned.



Lab: RIP Routing



Objective: Add the RIP routing protocol to your configuration.

Step 1 Configure RIP on all connected networks.

Step 2 Turn on RIP debugging by typing **debug ip rip**.

Step 3 Wait for routing updates and examine the results.

Step 4 Turn on RIP debugging by typing **debug ip rip** and examine the results.

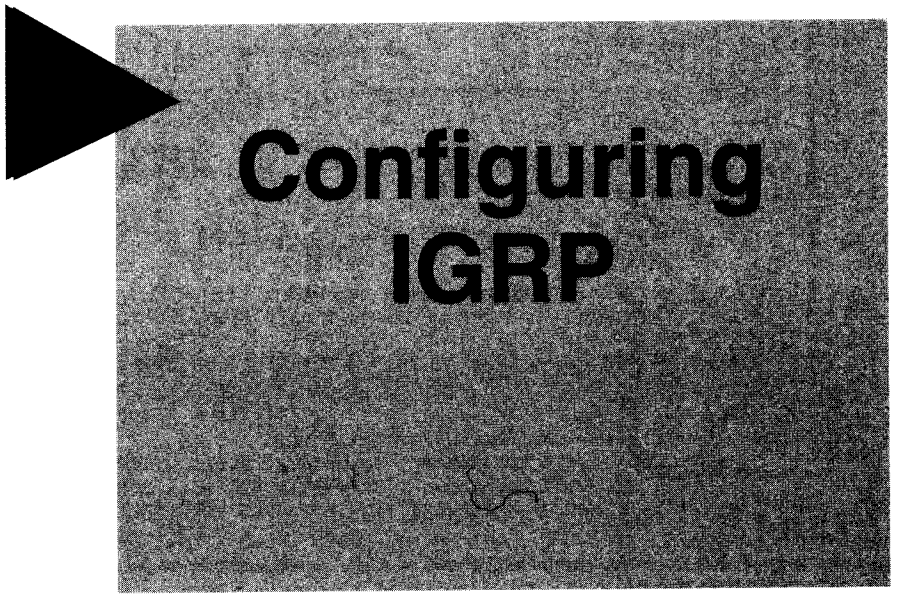
Step 5 Attempt to **telnet** to the other routers in your group.

Step 6 Use the **no shut** command to remove the disconnections to network 3.0.0.0.

Step 7 Use the **show ip route** command to examine the routing table.

Step 8 Save the configuration in nonvolatile memory.

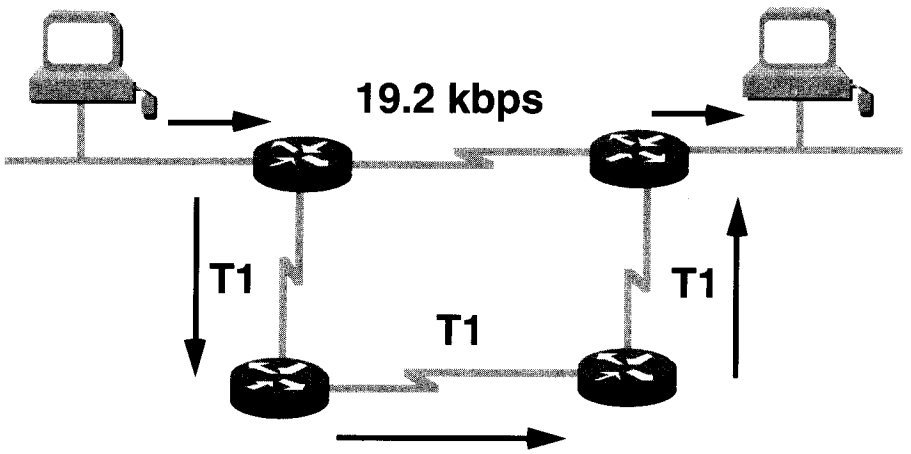
Step 9 Turn off RIP debugging.



26

Configuring IGRP

IGRP Overview



- **Composite metric selects the path**
- **Speed is the primary consideration**

27

IGRP is a distance vector routing protocol developed by Cisco. IGRP sends routing updates at 90-second intervals that advertise networks for a particular autonomous system.

The following are some key characteristics of IGRP:

- Design emphasizes:
  - Versatility to automatically handle indefinite, complex topologies.
  - Flexibility for segments having different bandwidth and delay characteristics.
  - Scalability to function in very large networks.

The IGRP routing protocol uses a combination of variables to determine a composite metric.

- Variables IGRP uses include:
  - Bandwidth
  - Delay
  - Load
  - Reliability
  - Maximum transmission unit (MTU)

IGRP Configuration

Router (config) #

router igrp *autonomous-system*

- **Defines IGRP as an IP routing process**

Router (config-router) #

network *network-number*

- **Selects participating attached networks**

28

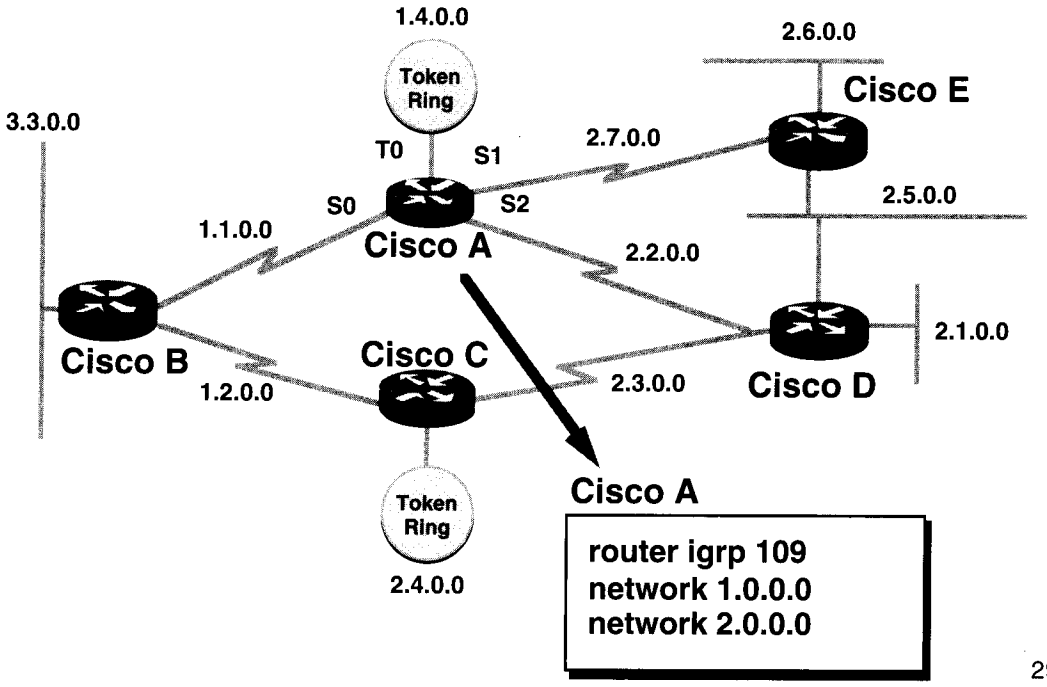
The **router igrp** command selects IGRP as a routing protocol.

router igrp Command	Description
autonomous-system	Identifies the IGRP router processes that will share routing information.

The **network** command specifies any directly connected networks to be included.

network Command	Description
network-number	Specifies a directly connected network: a NIC network number, not a subnet number or individual address.

IGRP Configuration Example



29

In the example:

- **router igrp 109**—Selects IGRP as the routing protocol for autonomous system 109.
- **network 1.0.0.0**—Specifies a directly connected network.
- **network 2.0.0.0**—Specifies a directly connected network.

IGRP is selected as the routing protocol for autonomous system 109. All interfaces connected to networks 1.0.0.0 and 2.0.0.0 will process IP traffic.

show ip protocol Command

```
Router> show ip protocol
Routing Protocol is "igrp 300"
  Sending updates every 90 seconds, next due in 55 seconds
  Invalid after 270 seconds, hold down 280, flushed after 630
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  IGRP maximum hopcount 100
  IGRP maximum metric variance 1
  Redistributing: igrp 300
  Routing for Networks:
    183.8.0.0
    144.253.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    144.253.100.1         100      0:00:52
    183.8.128.12          100      0:00:43
    183.8.64.130          100      0:01:02
  Distance:  (default is 100)
--More--
```

30

The **show ip protocol** command displays parameters, filters, and network information about the entire router.

The algorithm used to calculate the routing metric for IGRP is also shown in this display. It defines the value of the K1 through K5 metrics and the maximum hop count.

DEFAULT DISTANCES

DIRECT	0
STATIC	1
EIGRP	90
IGRP	100
OSPF	110
RIP	120

show ip interfaces Command

```
Router> show ip interfaces
Ethernet0 is up, line protocol is up
  Internet address is 183.8.128.2, subnet mask is 255.255.255.128
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is enabled
  Outgoing access list is not set
  Inbound  access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is enabled
  IP fast switching on the same interface is disabled
  IP SSE switching is disabled
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  Probe proxy name replies are disabled
 --More--
```

31

The **show ip interfaces** command displays the status and global parameters associated with an interface.

The Cisco IOS software automatically enters a directly connected route in the routing table if the interface is one through which software can send and receive packets. Such an interface is marked "up." If the interface is unusable, it is removed from the routing table. Removing the entry allows implementation of backup routes, if they exist.

show ip route Command

```
Router> show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default

Gateway of last resort is not set

     144.253.0.0 is subnetted (mask is 255.255.255.0), 1 subnets
C       144.253.100.0 is directly connected, Ethernet1
I    133.3.0.0 [100/1200] via 144.253.100.200, 00:00:57, Ethernet1
I    153.50.0.0 [100/1200] via 183.8.128.12, 00:00:05, Ethernet0
     183.8.0.0 is subnetted (mask is 255.255.255.128), 4 subnets
I       183.8.0.128 [100/180671] via 183.8.64.130, 00:00:27, Serial1
                    [100/180671] via 183.8.128.130, 00:00:27, Serial0
C       183.8.128.0 is directly connected, Ethernet0
C       183.8.64.128 is directly connected, Serial1
C       183.8.128.128 is directly connected, Serial0
I    172.16.0.0 [100/1200] via 144.253.100.1, 00:00:55, Ethernet1
I    192.3.63.0 [100/1300] via 144.253.100.200, 00:00:58, Ethernet1
```

NETWORK NEXT HOP LOCAL PORT

32

The **show ip route** command displays the contents of an IP routing table. The table contains a list of all known networks and subnets and the metrics associated with each entry.

Note that in this example the information was derived from IGRP or from direct connections.

debug ip rip Command

```
Router# debug ip rip
RIP protocol debugging is on
Router#
RIP: received update from 183.8.128.130 on Serial0
     183.8.0.128 in 1 hops
     183.8.64.128 in 1 hops
     0.0.0.0 in 16 hops (inaccessible)
RIP: received update from 183.8.64.130 on Serial1
     183.8.0.128 in 1 hops
     183.8.128.128 in 1 hops
     0.0.0.0 in 16 hops (inaccessible)
RIP: received update from 183.8.128.130 on Serial0
     183.8.0.128 in 1 hops
     183.8.64.128 in 1 hops
     0.0.0.0 in 16 hops (inaccessible)
RIP: sending update to 255.255.255.255 via Ethernet0 (183.8.128.2)
     subnet 183.8.0.128, metric 2
     subnet 183.8.64.128, metric 1
     subnet 183.8.128.128, metric 1
     default 0.0.0.0, metric 16
     network 144.253.0.0, metric 1
RIP: sending update to 255.255.255.255 via Ethernet1 (144.253.100.202)
     default 0.0.0.0, metric 16
     network 153.50.0.0, metric 2
     network 183.8.0.0, metric 1
```

33

The **debug ip rip** command displays RIP routing updates as they are sent and received.

In this example, the update is sent by 183.8.128.130. It reported on three routers, one of which is inaccessible because its hop count is greater than 15.

Updates were then broadcast through 183.8.128.2.

Summary

Routers can be configured to use one or more IP routing protocols

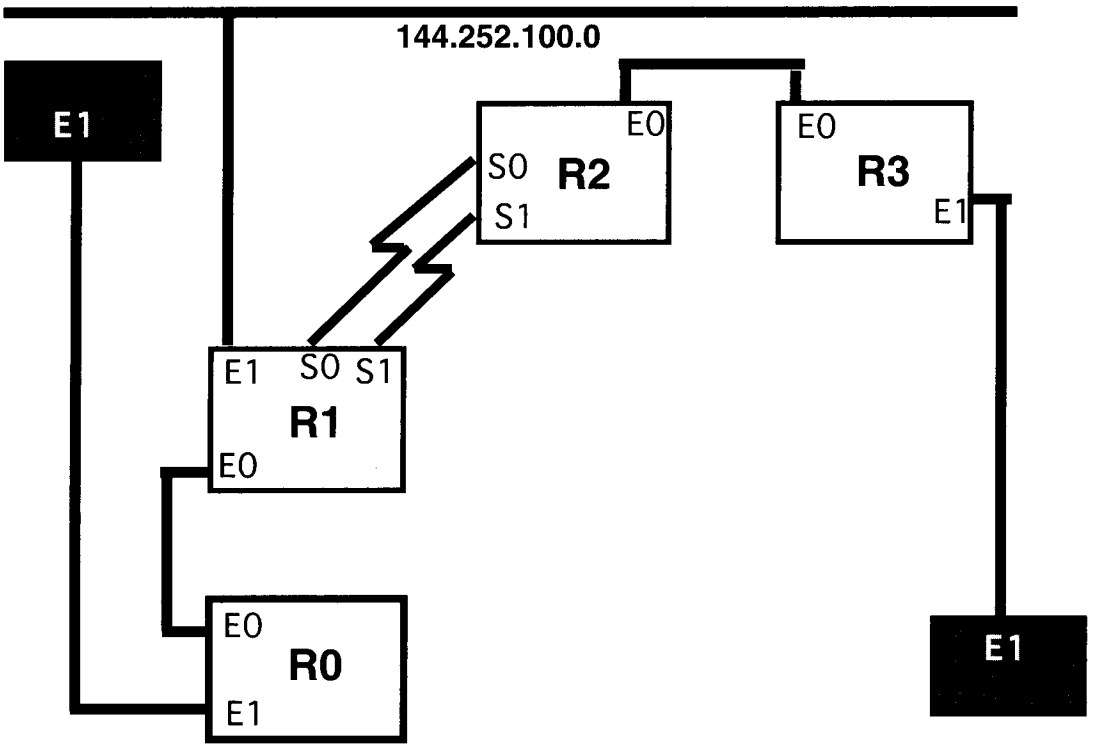
Two IP routing protocols are:

- **RIP**
- **IGRP**

34

Lab: IP Planning and Implementation

Classroom IP Planning Data Sheet



Objective: Add the IGRP routing protocol to your configuration.

Instructions: Plan for a classroom-wide IP internetwork. Refer to this data sheet and any other instructions from your instructor. Use the assigned address and bits of subnetting for your own workgroup as shown. Write the assigned address and bits of subnetting on the IP Planning Worksheet.

Workgroup	Assigned Address	Bits of Subnet
A	218.100.102.0	3
B	132.2.0.0	6
C	152.50.0.0	7
D	182.8.0.0	9
E	192.2.63.0	4
F	202.222.5.0	5

Use the addressing and bits of subnetting to connect to other groups as shown.

Location	E1 Address	Bits of Subnet
Tokyo	144.252.100.200	8
Bellevue	144.252.100.201	8
Dallas	144.252.100.202	8
Boston	144.252.100.203	8
Montreal	144.252.100.204	8
London	144.252.100.205	8

All groups will be in autonomous system 200.

Classroom IP Planning

Instructions: Given assigned network and bits of subnetting, plan for subnets and host addresses in your group. Fill in the IP Planning Worksheet and make sure all addresses are unique and legal.

Step 1 The previous page shows the IP addresses and the bits of subnet mask assigned by the authority. Assign a subnet and host address to each interface except those on E1. Work with the other students in your group. Once you get consensus on the entries for your group, write these addresses entries in the worksheet table.

Step 2 For the backbone address (E1 on R1) write the address entries derived from the address and subnet bits listed on the previous page into the worksheet.

Step 3 For E1 on R2, determine two unique and legal IP addresses on a subnet. These addresses must use a subnet in the range of subnets for your group. Enter one of these host addresses into the worksheet. Give the other address and subnet mask to the neighbor workgroup connected to E1 on R2.

Step 4 For E1 on R0, receive the IP host address and subnet mask from the neighbor workgroup on that data link and enter this address into the worksheet for E1 on R0.

Step 5 Check your results with the other students in your group. When you believe all entries are unique and legal, have your instructor review the worksheet for your group. Make any necessary corrections.

Step 6 When done, all members in your group should have the same worksheet entries. Do not do the next lab until directed to do so by the instructor.

IP Planning Worksheet

Group: _____ Assigned Address:_____ Subnet Mask:_____

Router R0 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
E1			

Router R1 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
E1			
S0			
S1			

Router R2 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
S0			
S1			

Router R3 host name:

Interface	IP Address	Subnet Mask	Subnet Number
E0			
E1			

Classroom IP Planning Notes

Classroom IP Implementation

Instructions: Using the work from the previous lab, we will form an internetwork within the training lab. This involves changing the IP addresses and converting the routing protocol to IGRP.

Step 1 Turn off RIP routing.

Step 2 Replace the IP address for each interface with the address from your IP Planning Worksheet. Use **configure** to change the running configuration. After you have entered all the new addresses, save this configuration to nonvolatile memory.

Step 3 Turn on the IGRP routing protocol; use the autonomous system number assigned—200.

Step 4 On the classroom white board, write the IP address for your Ethernet 0 interface. Make sure that this information is accurate because everyone in the room will be using this address for this lab and future labs.

Step 5 Use the **show ip route** command to look at the routing table in the routers.

Step 6 Use the **show cdp neighbor detail** command if necessary to obtain the interface addresses of your immediate workgroup neighbors.

Step 7 Use **telnet** and **ping** to test connectivity with your work group. If necessary, **shut** your group's E1 connections to other groups until you have connectivity within your own network.

Step 8 Check connectivity to every router in the class/lab internetwork. Use the addresses on the white board to **telnet** to the other routers. Record the Ethernet 0 address for all routers.

Group	Router Name	Ethernet 0 IP Address	OK
A	R0—Kyoto		
A	R1—Tokyo		
A	R2—Seattle		
A	R3—Tacoma		
B	R0—Olympia		
B	R1—Bellevue		
B	R2—Houston		
B	R3—Amarillo		
C	R0—Lubbock		
C	R1—Dallas		
C	R2—Chicago		
C	R3—Fargo		
D	R0—Nashua		
D	R1—Boston		
D	R2—Hartford		
D	R3—Lowell		
E	R0—Ottawa		
E	R1—Montreal		
E	R2—Quebec		
E	R3—Toronto		
F	R0—York		

Group	Router Name	Ethernet 0 IP Address	OK
F	R1—London		
F	R2—Paris		
F	R3—Rome		

Step 9 After you confirm that connectivity is established to an E0 addresses, indicate the table entry is OK. Save the configuration of your router in nonvolatile memory.

Optional and Extra IP Implementation Steps

Step 10 Optional: Help with troubleshooting classroom connectivity problems.

Use **ping**, **trace**, **telnet**, **show ip route**, **writer term**, and **debug** to identify, isolate, and repair problems in the network.

- If you discover a configuration problem, notify the responsible network administrator.

Note Do not in any way change the configuration of another student's router unless that student or the instructor gives you clear and explicit permission to do so.

- If you find that you must change the address on an interface, if it is the E0 address, make sure you update the white board entry so that others have an accurate address to use. Save the corrected configuration of your router in nonvolatile memory.

Step 11 Extra: Add a host table to your router configuration.

Use the classroom white board entries to set up a host table using each router's city name as a locally significant alias for that router's E0 IP address. Save the configuration of your router in nonvolatile memory.

Step 12 Extra: Add a banner to your configuration.

Use **banner motd** to enter a brief message that students will see when they get started from the console or Telnet as a virtual terminal connection. Test the banner yourself and modify as necessary. Save the configuration of your router in nonvolatile memory.

CHAPTER 13

Configuring Novell IPX

Objectives

Upon completion of this chapter, you will be able to perform the following tasks:

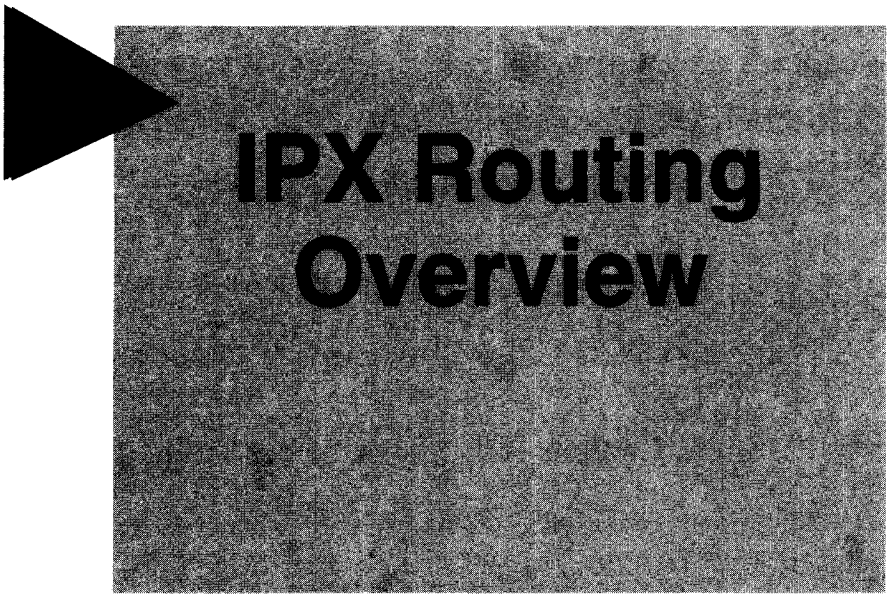
- **Describe the Novell IPX protocol stack**
- **Describe key features of Novell IPX**
- **List the required IPX address and encapsulation type**
- **Enable the Novell IPX protocol and configure interfaces**
- **Monitor Novell IPX operation on the router**

2

This chapter discusses configuring Cisco routers in Novell NetWare environments. It includes information about the Novell IPX protocol and about how to configure, verify, and monitor IPX routing.

Sections:

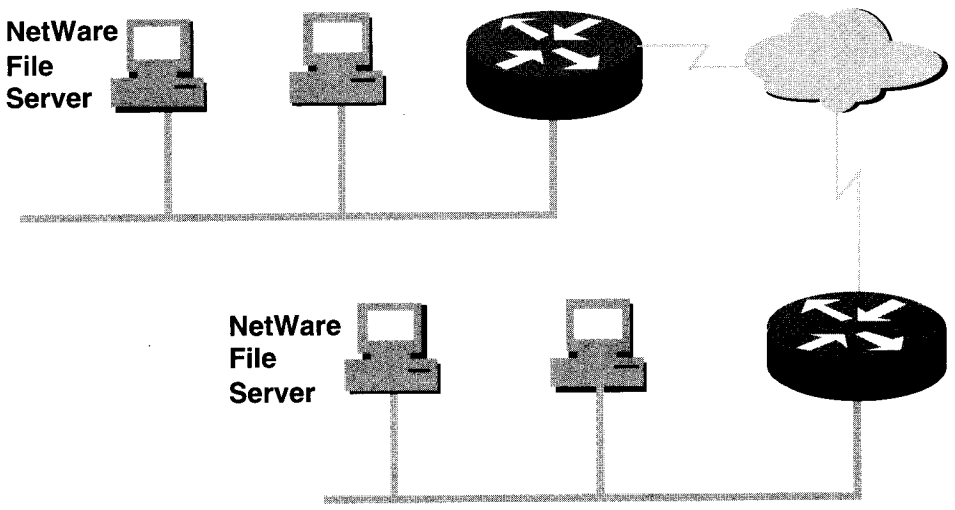
- IPX Routing Overview
- Configuring IPX Routing
- Verifying and Monitoring IPX Routing
- Answers to Exercise



3

IPX Routing Overview

Cisco Routers in NetWare Networks



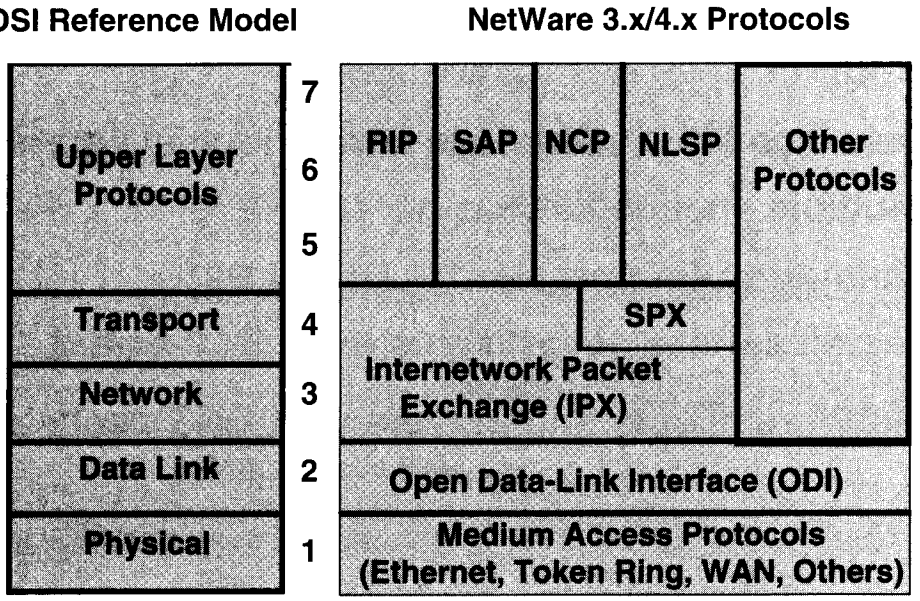
4

In today's networking environment, no one manufacturer can provide all the hardware and software required to support the computing needs of a business. As a result, most networks include a variety of vendor products, each one chosen for the powerful features it provides. For that reason, Cisco routers are often found in NetWare networks even though Novell offers routing products.

Cisco's routers offer these features that strengthen a Novell network configuration:

- Scalable routing and tunneling
- Multiple prioritization schemes and criteria
- Dial-on-demand and native ISDN support
- Native ATM support
- Performance
- NetWare IPX client support
- Rich diagnostic and troubleshooting features

Novell IPX Protocol Stack



5

Novell's Internetwork Packet Exchange (IPX) is a proprietary protocol derived from the Xerox Network Systems (XNS) protocol. Novell IPX is a:

- Datagram, connectionless protocol that does not require an acknowledgment for each packet.
- Layer 3 protocol that defines the internetwork and internode addresses.
- Router specification used to identify the Novell NetWare protocol suite.

Novell IPX uses:

- Routing Information Protocol (RIP) to facilitate the exchange of routing information.
- Proprietary Service Advertisement Protocol (SAP) to advertise network services.
- NetWare Core Protocol (NCP) to provide client-to-server connections and applications.
- Sequenced Packet Exchange (SPX) service for Layer 4 connection-oriented services.

Note Novell has introduced a link-state routing protocol called NetWare Link Services Protocol (NLSP). Novell intends for this new Layer 3 protocol to eventually replace RIP and SAP.

The NetWare protocol stack is compatible with Open Data-Link Interface (ODI) and all common media access protocols.

NLSP CAN REPLACE RIP & SAP

NLSP ONLY UPDATES AS REQUIRED

Key Novell IPX Features

- **Address is 80 bits (network.node)**
- **Interface MAC address is part of logical address**
- **Multiple encapsulations per interface**
- **Default routing protocol is Novell RIP**
- **Novell service advertisements in SAP traffic**
- **NetWare clients find servers with GNS packets**

6

A Novell IPX address has 80 bits: 32 bits for the network number and 48 bits for the node number.

The node number contains the MAC address of an interface.

Novell IPX supports multiple logical networks on an individual interface, and each network requires a single encapsulation type.

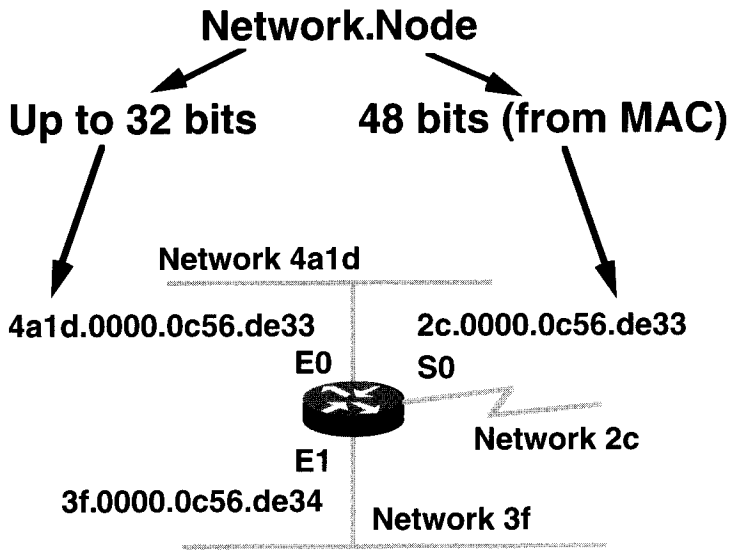
Novell RIP is the default routing protocol. Novell has also developed a link-state routing protocol, NLSP.

The client/server relationship is enhanced in IPX because servers automatically use the SAP protocol to advertise the services they provide.

One type of SAP advertisement is Get Nearest Server (GNS), which enables a client to locate the nearest server for login.

These features are discussed in detail in this chapter.

Novell IPX Addressing



- **Each interface has a unique address**

7

Novell IPX addressing uses a two-part address, the network number and the node number.

The IPX network number can be up to 16 hexadecimal digits in length. This number is assigned by the network administrator.

The example features the IPX network 4a1d. Other IPX networks shown are 2c and 3f.

The IPX node number is 12 hexadecimal digits in length. This number is usually the MAC address obtained from a router interface that has a MAC address.

The example features the IPX node 0000.0c56.de33. Another node address is 0000.0c56.de34.

Notice that the same node number appears for both E0 and S0. Serial interfaces do not have MAC addresses so Novell IPX obtained this node number for S0 by using the MAC address from E0.

Each interface retains its own address. The use of the MAC address in the logical address eliminates the need for an Address Resolution Protocol (ARP).

How to Determine the IPX Address

- **Ask the NetWare administator**

- **Use a Cisco IOS command to check on the neighbor Cisco router**

- **Use NetWare command to check on the NetWare file server/router**

8

You must use a valid IPX network address when you configure the Cisco router. Because the Novell NetWare networks are likely to be already established with IPX addresses, determine the IPX address from these already established networks. The IPX network address refers to the "wire"; all routers on the same wire must share the same IPX network address.

The first and recommended way to find out what address to use is to ask the NetWare administrator. Make sure that the NetWare administrator specifies the IPX network address for the same network where you want to enable IPX on your Cisco router. The Cisco router must use the same network as the NetWare file server (or other source of the address) specified by the NetWare administrator.

If you cannot obtain an IPX address to use from the NetWare administrator, you can get the neighbor's IPX address directly from a neighbor router. Pick the most appropriate of the several methods available to do this:

- If the neighbor router is another Cisco router, you can use a Cisco IOS command to show Cisco Discovery Protocol (CDP) neighbor details.
- You can Telnet to the neighbor router, enter the appropriate mode, then display the running configuration on the neighbor.
- If the neighbor router is not a Cisco router (is a NetWare PC-based router, or a NetWare file server), you may be able to attach or log in and use the NetWare utility **config** to determine the address.

On the Cisco router, you must use the same IPX network address as the address that already exists on that network.

Multiple Novell Encapsulations

For example, four types of Ethernet framing

Novell IPX Name	Framing Structure
• Ethernet_II	Ethernet | IPX
• Ethernet_802.2 (default for NetWare 3.12 or later)	802.3 | 802.2 LLC | IPX
• Ethernet_SNAP	802.3 | 802.2 LLC | SNAP | IPX
• Ethernet_802.3 (default for NetWare 3.11 or earlier)	802.3 | IPX

9

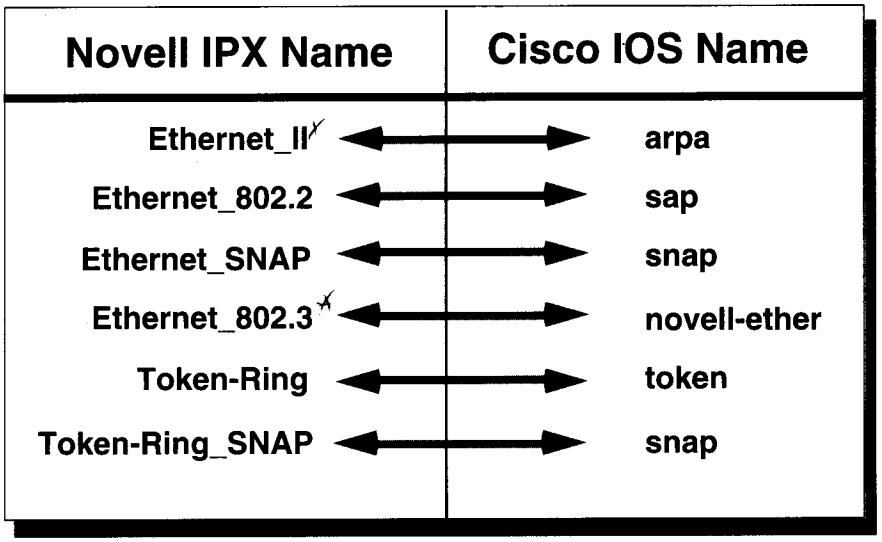
The Novell IPX protocol on Cisco routers supports all the framing variations used on Novell NetWare implementations. These framing types include service access point (SAP), Ethernet, 802.3 with 802.2 logical link control (LLC) protocol, and Subnetwork Access Protocol (SNAP).

There are four different Ethernet framing types with variations in the fields they use. Each encapsulation type is appropriate in specific situations:

- Ethernet II—Used with TCP/IP and DECnet.
- Ethernet 802.2—Used with NetWare 4.*x* and OSI routing.
- Ethernet SNAP—Used with TCP/IP and AppleTalk.
- Ethernet 802.3—Also called raw Ethernet; used with early NetWare versions 2.*x* and 3.*x*.

Note Multiple encapsulations can be specified on an interface, but only if multiple network numbers have also been assigned. Although several encapsulation types can share the same interface, clients and servers with different encapsulation types cannot communicate directly with each other.

Cisco Encapsulation Names



- **Specify encapsulation when you configure IPX network**

10

When you configure an IPX network, you may need to specify a nondefault encapsulation type. To help you specify the appropriate encapsulation type, use the table in the graphic. The table matches the Novell framing terms to equivalent Cisco IOS names for the same framing types.

When you configure Cisco IOS software for Novell IPX, use the Cisco name for the appropriate encapsulation.

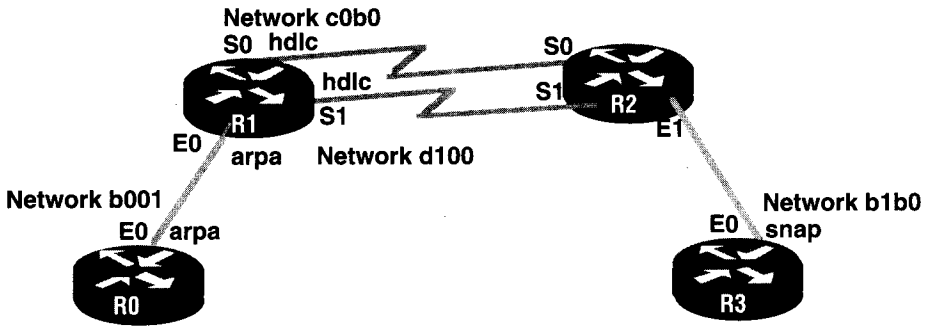
If you do not specify an encapsulation type when you configure the router for IPX, the router will use a default encapsulation type on its interfaces.

The default encapsulation types on Cisco router interfaces and their keywords are:

- Ethernet—**novell-ether**
- Token Ring—**snap**
- FDDI—**snap**

Exercise: IPX Parameter Planning

R2 Interface Name	Network Address	Encapsulation
S0	c0b0	HDLC
S1	d100	HDLC
E1	b1b0	SNAP



- **Write the IPX addresses and encapsulation types for R2** 11

Exercise: IPX Parameter Planning

Objective: List the required IPX address and encapsulation type.

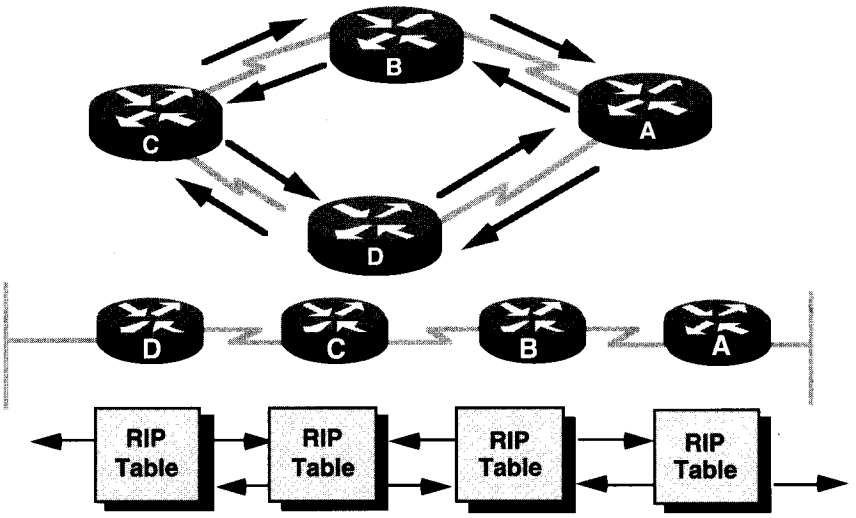
You will be solving for router R2. In the graphic, four routers will be configured to run Novell IPX. You are given crucial IPX details necessary for configuration for only three of the four routers. This information is summarized in the following table:

Router Name	Interface Name	IPX Network Address	Encapsulation Type
R0	E0	b001	arpa
R1	S0	c0b0	hdlc
	E0	b001	arpa
	S1	d100	hdlc
R3	E0	b1b0	snap

Your task is to determine the IPX network addresses and encapsulation types to use when you configure the R2 router.

Write your answers in the rectangle in the graphic.

RIP—The IPX Routing Protocol



- **Uses ticks (about 1/18 sec.) and hop count (maximum of 15 hops)**
- **Broadcasts routing information to neighbor routers every 60 seconds**

12

Novell RIP is a distance vector routing protocol. RIP has two metrics: ticks and hops. Ticks (a time measure) and hop count (a count of each router traversed) are the IPX metrics for path decisions.

RIP checks its two distance vector metrics by first comparing the ticks for path alternatives. If two or more paths have the same tick value, RIP compares the hop count. If two or more paths have the same hop count, the router will try to use a user-defined tiebreaker.

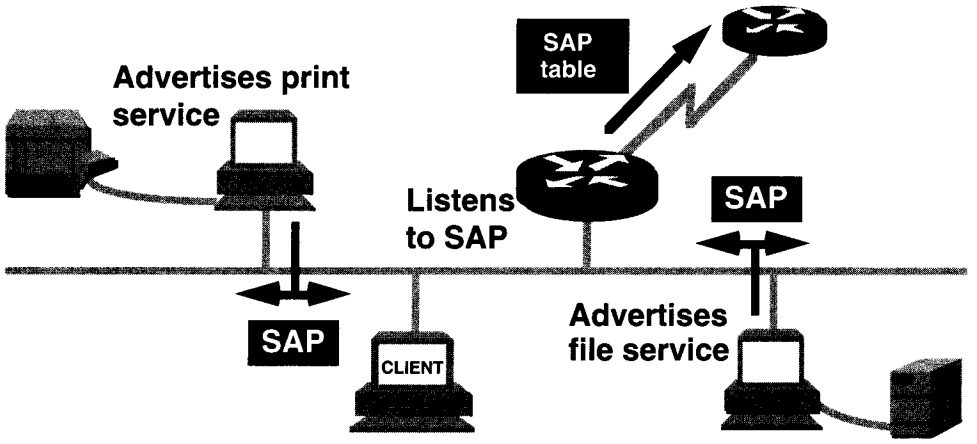
Each IPX router passes periodic copies of its RIP routing table to its direct neighbor IPX routers. The neighbor IPX routers add distance vectors as required before passing copies of their RIP tables to their own neighbors.

A "best information" split-horizon algorithm prevents the neighbor from broadcasting RIP tables about IPX information back to the networks from where it received that information.

RIP also uses an information aging mechanism to handle conditions where an IPX router goes down without any explicit message to its neighbors. Periodic updates reset the aging timer.

Routing table updates are sent at 60-second intervals. This update frequency can cause excessive overhead traffic on some internetworks.

SAP—Service Advertisements



- **SAP packets advertise all NetWare network services**
- **Can add excessive broadcast traffic to the network**

13

All the servers on NetWare internetworks can advertise their services and addresses. All versions of NetWare support SAP broadcasts to locate registered network services. Adding, finding, and removing services on the internetwork is dynamic because of SAP advertisements.

Each SAP service is an object type identified by a hexadecimal number. Examples:

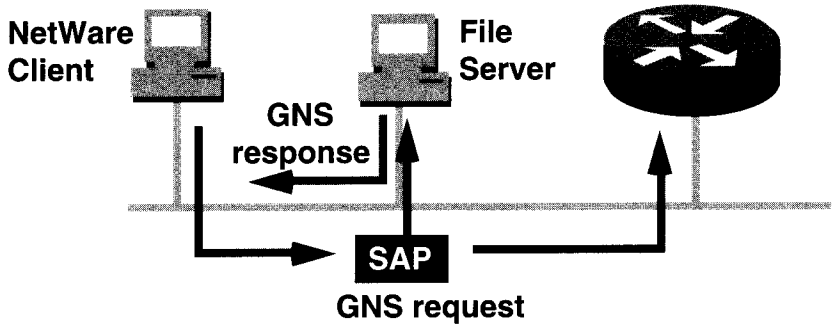
4	NetWare file server
7	Print server
24	Remote bridge server (router)

All servers and routers keep a complete list of the services available throughout the network in server information tables. Like RIP, SAP also uses an aging mechanism to identify and remove table entries that become invalid.

By default, service advertisements occur at 60-second intervals. However, although this might work well on a LAN, this advertisement can require too much bandwidth to be acceptable on large internetworks, or in internetworks linked on WAN serial connections.

Routers do not forward SAP broadcasts. Instead, each router builds its own SAP table and forwards the SAP table to other routers. By default this occurs every 60 seconds.

GNS—Clients Get Nearest Server



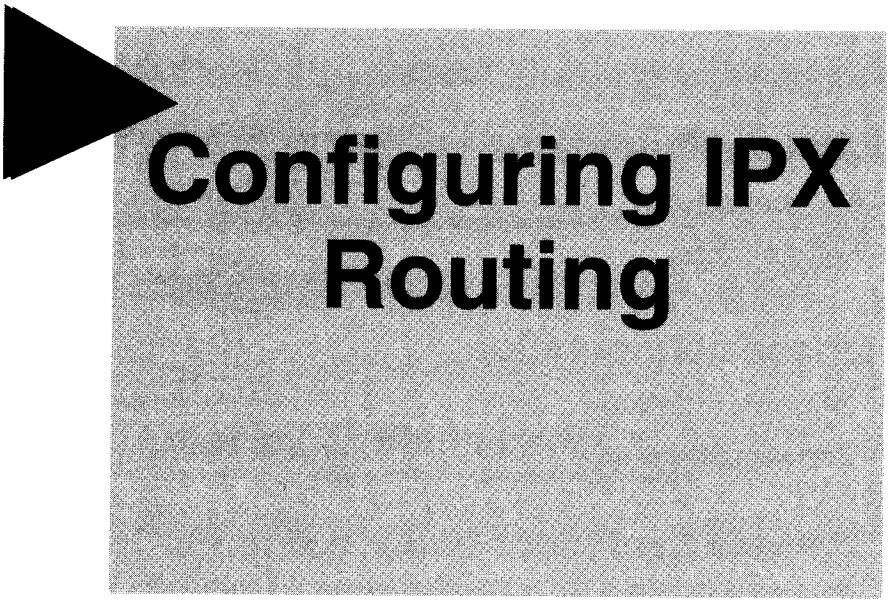
- **GNS is a broadcast from a client needing a server**
- **File server and Cisco router get this SAP packet**
- **File server provides GNS response**

14

The IPX client/server interaction begins when the client powers up and runs its client startup programs. These programs use the client's network adapter on the LAN and initiate the connection sequence for the NetWare shell to use.

The Get Nearest Server (GNS) is a broadcast that comes from a client using IPX SAP. The nearest NetWare file server responds with another SAP; the protocol type is Give Nearest Server. From that point on, the client can log in to the target server, make a connection, set the packet size, and proceed to use server resources.

If a NetWare server is located on the segment, it will respond to the client request. The Cisco router will not respond to the GNS request. If there are no NetWare files, the Cisco router will respond to a GNS request on a network segment.

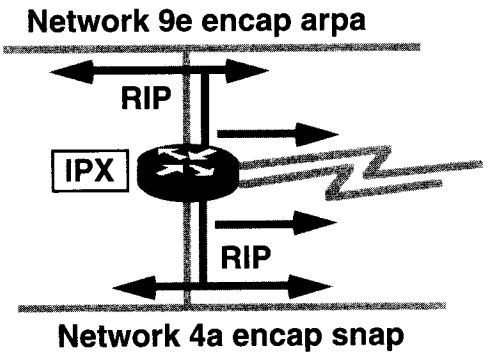


15

Configuring IPX Routing

Novell IPX Configuration Tasks

- **Global configuration**
  - **IPX routing**
  - **Load sharing**
- **Interface configuration**
  - **Network numbers**
  - **Encapsulation type**



16

Configuration of Novell IPX as a routing protocol involves both global and interface parameters.

Global tasks:

- Start the IPX routing process.
- Enable load sharing if appropriate for your network. Load sharing is the division of routing tasks evenly among multiple routers to balance the work and improve network performance.

Interface tasks:

- Assign unique network numbers to each interface. Multiple network numbers can be assigned to an interface, allowing support of different encapsulation types.
- Set the optional encapsulation type if it is different from the default.

Novell IPX Global Configuration

Router (config) #

ipx routing [*node*]

- **Enables Novell IPX routing**

Router (config) #

ipx maximum-paths *paths*

- **Configures round-robin load sharing over multiple equal metric paths**

17

The **ipx routing** command enables Novell IPX routing. If no node address is specified, the Cisco router uses the MAC address of the interface.

If a Cisco router has only serial interfaces, an address must be specified.

The **ipx maximum-paths** command enables load sharing.

ipx maximum-paths Command	Description
paths	Maximum number of parallel paths to the destination; the default is 1 and the maximum is 512.

Novell IPX Interface Configuration

Router (config-if) #

ipx network ***number*** [**encapsulation** ***encapsulation-type***] [**secondary**]

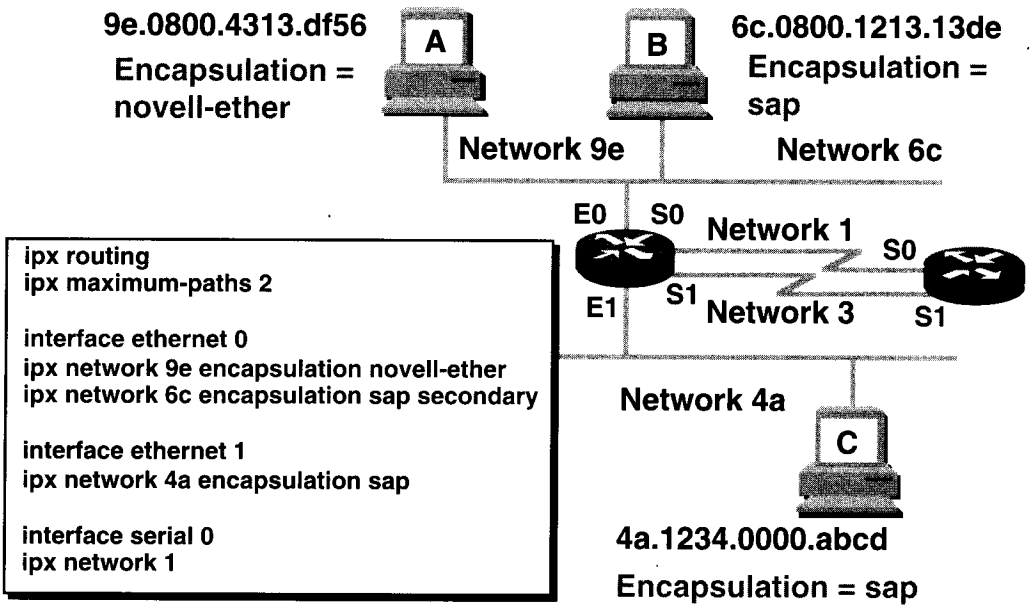
- **Assigns primary and secondary network number and encapsulation**

The **ipx network** command enables Novell IPX processing on this interface.

ipx network Command	Description
number	Each interface must have a unique Novell IPX network number that is specified in hexadecimal and up to eight hexadecimal numbers in length.
encapsulation -type	(Optional) Specifies the encapsulation type for the interface. Can be one of the following encapsulation types: **novell-ether**, **sap**, **arpa**, **snap**.
secondary	(Optional) Applies another network number and encapsulation to the interface.

Assigning the second network number is necessary if an additional encapsulation type is linked to an individual network.

Novell IPX Configuration Example



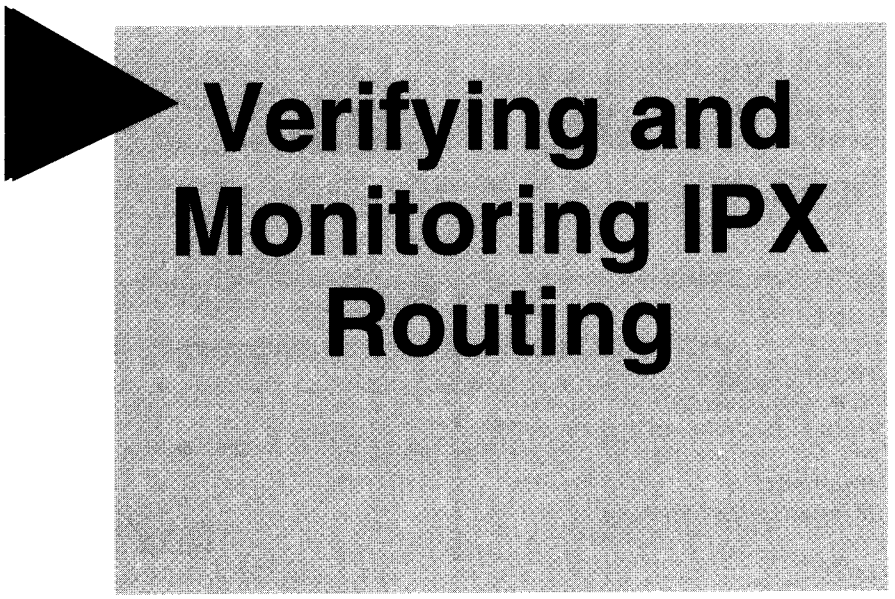
19

In the example:

Command	Description
ipx routing	Selects IPX as a routing protocol and starts the routing process.
ipx maximum-paths 2	Allows load sharing over parallel metric paths to the destination. The number of parallel paths used is limited to two.

Command	Description
ipx network 9e encapsulation novell-ether	
9e	Network number assigned to interface E0.
encapsulation novell-ether	Specifies that Novell's unique frame format is used on this network segment. Cisco's keyword is **novell-ether**; Novell's terminology is Ethernet_802.3.

Command	Description
ipx network 6c encapsulation sap secondary	
6c	Assigns a secondary network number to E0.
encapsulation sap secondary	Specifies the encapsulation type for this secondary network on E0. Cisco's keyword is **sap**; Novell's terminology is Ethernet_802.2. The type frame is Ethernet 802.3 with 802.2 LLC included.



20

Verifying and Monitoring IPX Routing

Verifying IPX Operation

Monitoring Commands	Troubleshooting Commands
show ipx interface	debug ipx routing activity
show ipx route	debug ipx sap
show ipx servers	
show ipx traffic	

21

Once IPX routing is configured, you can monitor and troubleshoot it using the following commands:

Monitoring Command	Displays
show ipx interface	IPX status and parameters.
show ipx route	Routing table contents.
show ipx servers	IPX server list.
show ipx traffic	Number and type of packets.

Troubleshooting Command	Displays
debug ipx routing activity	Information about RIP update packets.
debug ipx sap	Information about SAP update packets.

Each of these commands is discussed in detail on the pages that follow.

Monitoring IPX Status

```
Router# show ipx interface ethernet 0
Ethernet0 is up, line protocol is up
  IPX address is 3010.aa00.0400.0284, NOVELL-ETHER [up] line-up, RIPPQ: 0, SAPPQ: 0
  Delay of this Novell network, in ticks is 1
  IPXWAN processing not enabled on this interface.
  IPX SAP update interval is 1 minute(s)
  IPX type 20 propagation packet forwarding is disabled
  Outgoing access list is not set
  IPX Helper access list is not set
  SAP Input filter list is not set
  SAP Output filter list is not set
  SAP Router filter list is not set
  SAP GNS output filter list is not set
  Input filter list is not set
  Output filter list is not set
  Router filter list is not set
  Netbios Input host access list is not set
  Netbios Input bytes access list is not set
  Netbios Output host access list is not set
  Netbios Output bytes access list is not set
  Update time is 60 seconds
  IPX accounting is disabled
  IPX fast switching is configured (enabled)
  IPX SSE switching is disabled
  RIP packets received 1, RIP packets sent 10006
  SAP packets received 1, SAP packets sent 6
--More--
```

22

The **show ipx interface** command shows the status of IPX interface and IPX parameters configured on each interface.

The first highlighted line shows the IPX address, the type of encapsulation, and the status of the interface.

The middle set of highlighting shows that the SAP filters are not set.

The last highlighted line shows that fast switching is enabled.

You can manually set the tick metric. Use the command **ipx delay** *number* where *number* is the ticks to associate with an interface. This command manually overrides the following defaults on the Cisco router:

- For LAN interfaces, one tick
- For WAN interfaces, six ticks

Monitoring IPX Routing Tables

```
Router# show ipx route
Codes: C - Connected primary network, c - Connected secondary network
       R - RIP, E - EIGRP, S - static, W - IPXWAN connected
5 Total IPX routes

Up to 2 parallel paths allowed       Novell routing algorithm variant in use

R  Net 3030 [6/1] via 3021.0000.0c03.13d3,  23 sec, 1 uses, Serial1
                  via 3020.0000.0c03.13d3,  23 sec, 0 uses, Serial0
C  Net 3020 (X25), is directly connected, 15 uses, Serial0
C  Net 3021 (HDLC), is directly connected, 15 uses, Serial1
C  Net 3010 (NOVELL-ETHER), is directly connected, 15 uses, Ethernet0
C  Net 3000 (NOVELL-ETHER), is directly connected, 15 uses, Ethernet1
```

Ticks/ Hops

23

The **show ipx route** command displays the contents of the IPX routing table.

The first highlighted line provides routing information for a remote network:

- The information was learned from a RIP update.
- The network is number 3030.
- It is located six ticks or one hop away. This information is used to determine best routes. If there is a tie between ticks, hops are used to break the tie.
- The next hop in the path is router 3021.0000.0c03.13d3.
- The information was updated 23 seconds ago.
- The updates will be sent through the interface named Serial1.

The second line of highlighting provides information about a direct connection:

- The network number is 3010.
- The encapsulation type is NOVELL-ETHER.

Monitoring IPX Servers List

```
Router> show ipx servers
Codes: P - Periodic, I - Incremental, H - Holddown, S - static
1 Total IPX Servers

Table ordering is based on routing and server info

Type     Name       Net     Address    Port     Route    Hops Itf
P4       MAXINE   AD33000.0000.1b04.0288:0451 332800/1  2    Et3
```

24

The **show ipx servers** command lists the IPX servers discovered through SAP advertisements.

This example provides the following information:

- The service learned about the server from a SAP update
- The server name, network location, device address, and source socket number
- The ticks and hops for the route (taken from the routing table)
- The number of hops (taken from the SAP protocol)
- The interface through which to reach the server

Monitoring IPX Traffic

```
Router# show ipx traffic
System Traffic for 2018.0000.0000.0001 System-Name: dtp-18
Rcvd:   23916 total, 13785 format errors, 0 checksum errors, 0 bad hop
count,
        0 packets pitched, 23916 local destination, 0 multicast
Bcast:  17111 received, 9486 sent
Sent:   16707 generated, 0 forwarded
        0 encapsulation failed, 0 no route
SAP:    6 SAP requests, 6 SAP replies, 2309 servers
        0 SAP Nearest Name requests, 0 replies
        0 SAP General Name requests, 0 replies
        1521 SAP advertisements received, 2212 sent
        0 SAP flash updates sent, 0 SAP format errors
RIP:    6 RIP requests, 6 RIP replies, 2979 routes
        8033 RIP advertisements received, 4300 sent
        154 RIP flash updates sent, 0 RIP format errors
Echo:   Rcvd 0 requests, 0 replies
        Sent 0 requests, 0 replies
        0 unknown: 0 no socket, 0 filtered, 0 no helper
        0 SAPs throttled, freed NDB len 0
Watchdog:
        0 packets received, 0 replies spoofed
Queue lengths:
        IPX input: 0, SAP 0, RIP 0, GNS 0
        SAP throttling length: 0/(no limit), 0 nets pending lost route
reply
        Delayed process creation: 0
```

25

The **show ipx traffic** command displays information about the number and type of IPX packets received and transmitted by the router.

Notice in this example that a high percentage of the total number of packets received and sent were RIP advertisements. This is because this sample was taken from a lab network with essentially no user traffic on it. This screen shows how much overhead traffic IPX generates.

Troubleshooting IPX Routing

```
Router# debug ipx routing activity
IPX routing debugging is on
Router#
IPXRIP: positing full update to 3010.ffff.ffff.ffff via Ethernet0 (broadcast)
IPXRIP: positing full update to 3000.ffff.ffff.ffff via Ethernet1 (broadcast)
IPXRIP: positing full update to 3020.ffff.ffff.ffff via Serial0 (broadcast)
IPXRIP: positing full update to 3021.ffff.ffff.ffff via Serial1 (broadcast)
IPXRIP: sending update to 3020.ffff.ffff.ffff via Serial0
IPXRIP: src=3020.0000.0c03.14d8, dst=3020.ffff.ffff.ffff, packet sent
    network 3021, hops 1,  delay 6
    network 3010, hops 1,  delay 6
    network 3000, hops 1,  delay 6
IPXRIP: sending update to 3021.ffff.ffff.ffff via Serial1
IPXRIP: src=3021.0000.0c03.14d8, dst=3021.ffff.ffff.ffff, packet sent
    network 3020, hops 1,  delay 6
    network 3010, hops 1,  delay 6
    network 3000, hops 1,  delay 6
IPXRIP: sending update to 3010.ffff.ffff.ffff via Ethernet0
IPXRIP: src=3010.aa00.0400.0284, dst=3010.ffff.ffff.ffff, packet sent
    network 3030, hops 2,  delay 7
    network 3020, hops 1,  delay 1
    network 3021, hops 1,  delay 1
    network 3000, hops 1,  delay 1
IPXRIP: sending update to 3000.ffff.ffff.ffff via Ethernet1
```

26

The **debug ipx routing activity** command displays information about IPX routing update packets that are transmitted or received.

A router sends an update every 60 seconds. Each update packet can contain up to 50 entries. If there are more than 50 entries in the routing table, the update will include more than one packet.

In this example, the router is sending updates but not receiving them. Updates received from other routers would also appear in this listing.

Troubleshooting IPX SAP

```
Router# debug ipx sap
IPX SAP debugging is on
Router#
NovellSAP: at 0023F778
I SAP Response type 0x2 len 160 src:160.0000.0c00.070d dest:160.ffff.ffff.ffff(452)
 type 0x4,  "HELLO2", 199.0002.0004.0006 (451), 2 hops
 type 0x4,  "HELLO1", 199.0002.0004.0008 (451), 2 hops
NovellSAP: sending update to 160
NovellSAP: at 00169080
 O SAP Update type 0x2 len 96 ssoc:0x452 dest:160.ffff.ffff.ffff(452)
 Novell: type 0x4  "Magnolia", 42.0000.0000,0001 (451), 2 hops
```


The **debug ipx sap** command displays information about IPX SAP packets that are transmitted or received.

Like RIP updates, these SAP updates are sent every 60 seconds and may contain multiple packets. Each SAP packet appears as multiple lines in the output, including a packet summary message and a service detail message.

SAP responses may be one of these types:

- 0x1—General query
- 0x2—General response
- 0x3—Get Nearest Server request
- 0x4—Get Nearest Server response

In each line, the address and distance of the responding or target router is listed.

Summary

Address is network.node

Logical address contains interface MAC address

IPX interface configuration supports multiple data-link encapsulations

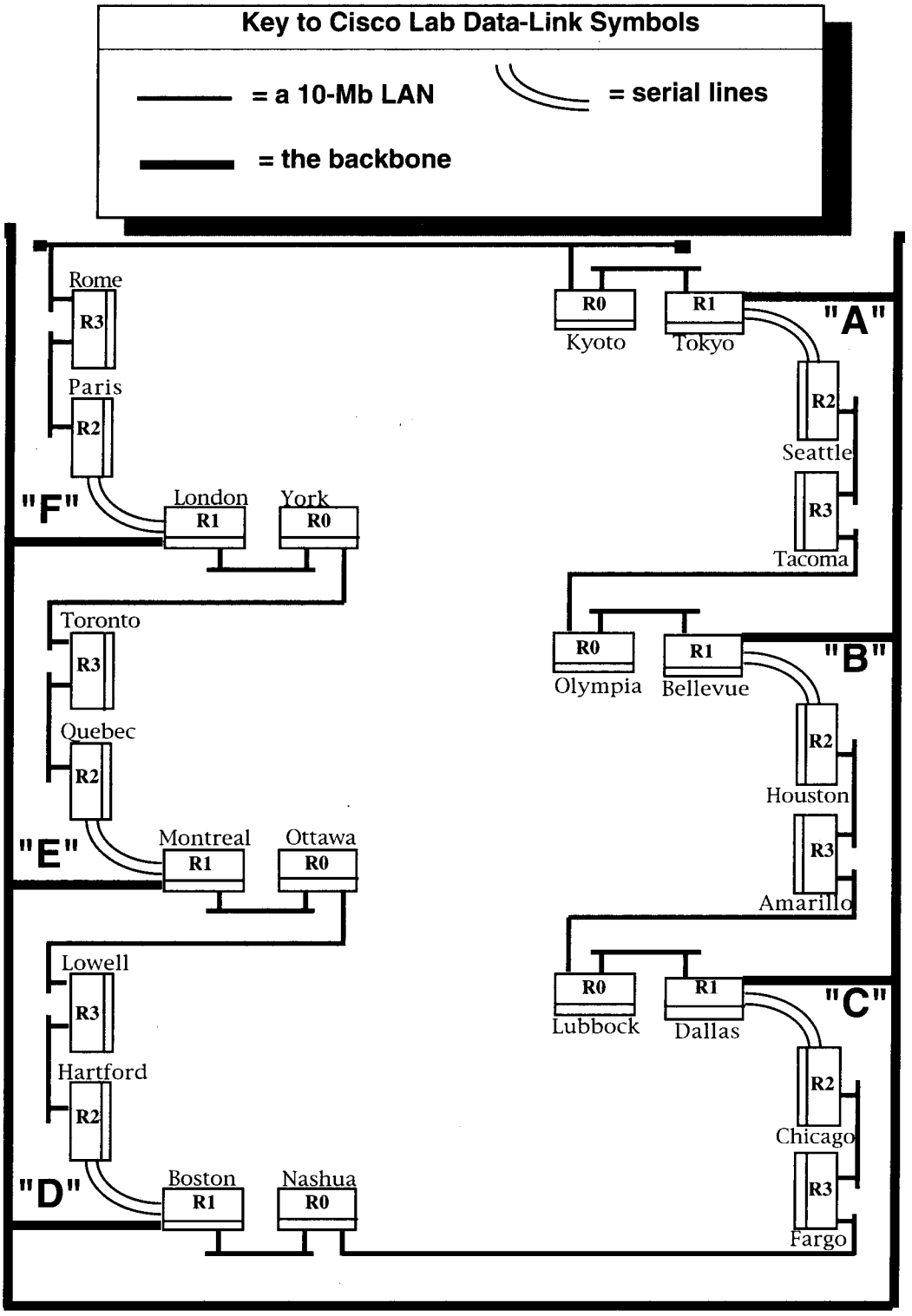
RIP uses the distance vectors of ticks and hops

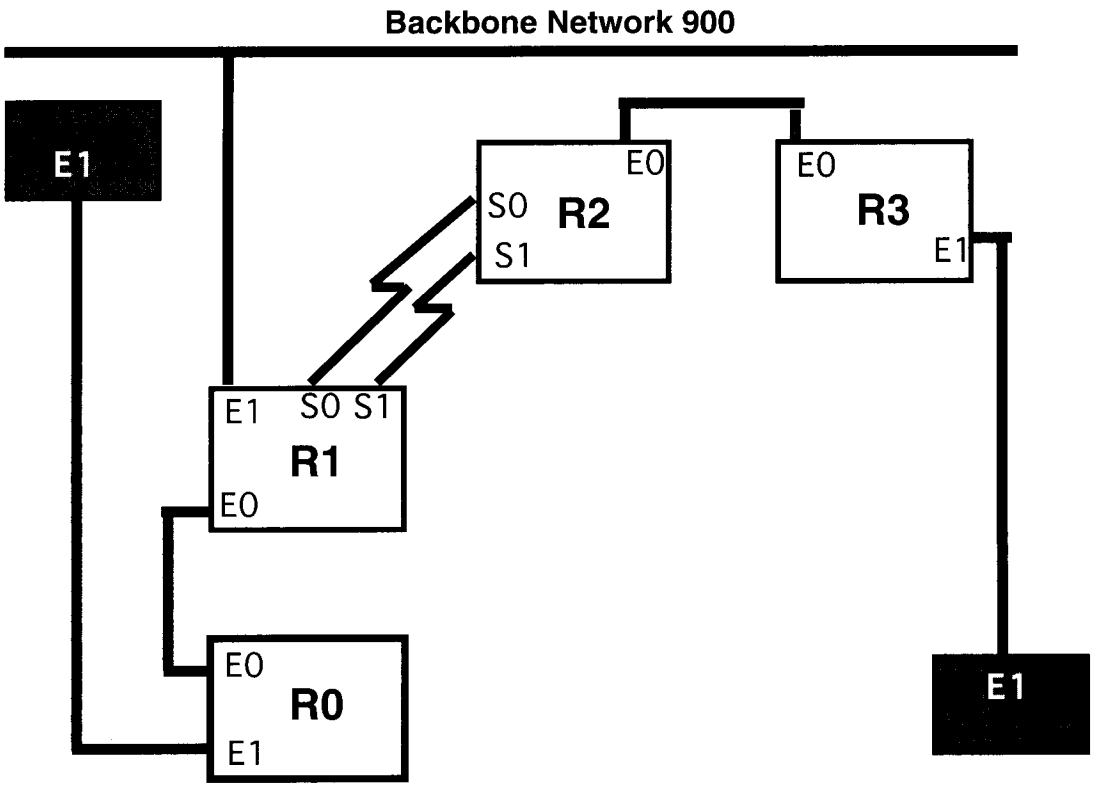
SAPs and GNS broadcasts function to connect clients and servers

28

Lab: IPX Planning and Implementation

Map of the Classroom/Lab Novell IPX Internetwork





Objective: Enable the Novell IPX protocol and configure interfaces.

Objective: Monitor Novell IPX operation on the router.

Instructions: Enable the Novell IPX protocol and configure interfaces. Then monitor Novell IPX operation on the router. Use the assigned IPX address range for your own workgroup as shown.

Workgroup	Address Range
A	1000-1999
B	2000-2999
C	3000-3999
D	4000-4999
E	5000-5999
F	6000-6999
Backbone	9000

IPX Implementation

Instructions: Using the work from the previous lab, we will form an IPX internetwork within the training lab. Configure Novell IPX and establish connectivity within your workgroup. Next, expand your network connectivity to incorporate the rest of the groups in the room.

Step 1 Shut down all E1 interfaces that connect your network to other groups.

Step 2 Using the **configure** command, start IPX routing.

Step 3 Assign the IPX network numbers to the interfaces.

For any serial interfaces, issue the command **ipx maximum-paths 2**.

Assign network numbers to the E1 interfaces, even though they are shut down at this time. The E1 network number for the backbone has been provided. The E1 intergroup connections were negotiated during the planning stage. The E1 interfaces will be reactivated in step 7.

Step 4 Use the **show ipx interface** command to verify address assignment.

Step 5 Use the **show ipx route** command to verify entries in the routing table.

Step 6 Use the **ping** command to verify connectivity across your workgroup. In order to get the address, you will need to **telnet** to the target router to find the IPX address.

Step 7 Check the network numbers assigned to the E1 interfaces. Reactivate E1 interfaces using the **no shutdown** command and expand your network to incorporate all networks in the room.

Step 8 Use the IP addresses of the E0 interface to **telnet** to routers in other groups in the network. (If you are on an R0 router, **telnet** to the R0 router in every other group. If you are on an R1 router, test connectivity to every R1 router in the room.

Step 9 Use the **show ipx interface** command to discover the IPX addresses on the remote router. Record them in the table.

Circle the routers you are testing: R0 R1 R2 R3

Group	IPX Address
A	
B	
C	
D	
E	
F	

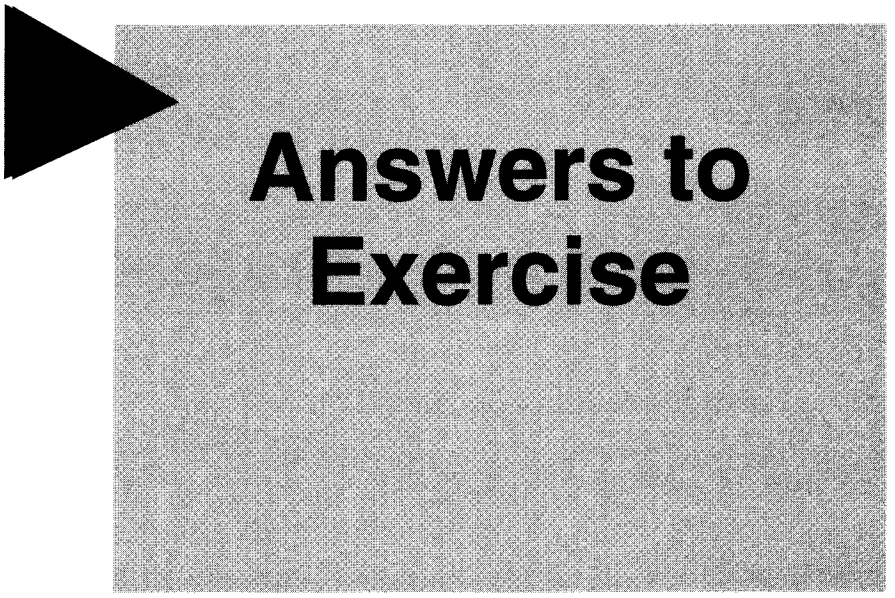
Step 10 Use the **ping ipx** command from your router to verify IPX connectivity across all workgroups.

Step 11 Once you have confirmed that connectivity has been established, save the configuration of your router in nonvolatile memory.

Optional and Extra IPX Implementation Steps

Step 12 Optional: Help with troubleshooting classroom connectivity problems.

- Use **ipx ping**, **show ipx route**, **show running-config,** and **debug** to identify, isolate, and repair problems in the network. If you discover a configuration problem, notify the responsible network administrator.



33

Answers to Exercise

Exercise: IPX Parameter Planning

R2 Interface Name	Network Address	Encapsulation
S0	c0b0	hdlc
S1	d100	hdlc
E1	b1b0	snap

CHAPTER 14

Configuring AppleTalk

Objectives

Upon completion of this chapter, you will be able to perform the following tasks:

Describe the AppleTalk protocol stack

Plan an AppleTalk network

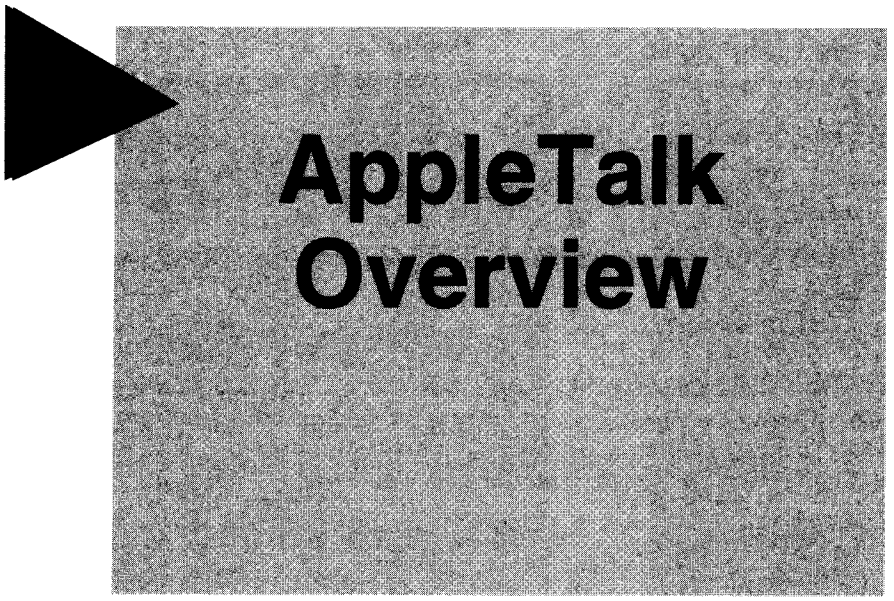
Enable AppleTalk protocol and configure AppleTalk interfaces

Monitor AppleTalk operation in the router

This chapter discusses how to configure AppleTalk routing. It presents information about the AppleTalk protocol stack and how AppleTalk addressing works. It explains how to configure and monitor AppleTalk routing.

Sections:

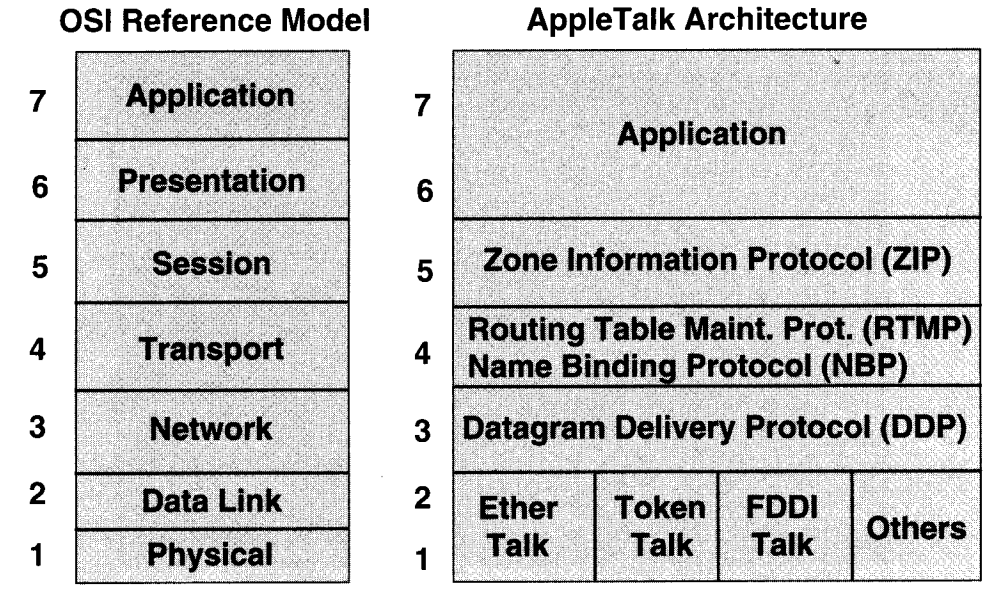
- AppleTalk Overview
- Configuring AppleTalk



3

AppleTalk Overview

AppleTalk Protocol Stack



4

At the hardware layers, most standard media types are supported. Many Apple products contain a LocalTalk interface that operates over twisted-pair cabling at 230 kbps. The LocalTalk interface is not available on Cisco products. LocalTalk devices can be adapted to Ethernet or other LAN environments.

At Layer 3 in the AppleTalk architecture, the Datagram Delivery Protocol (DDP) provides a connectionless datagram service.

At Layer 4 in the AppleTalk architecture, the Name Binding Protocol (NBP) provides name-to-address association. Routing table content is provided by the Routing Table Maintenance Protocol (RTMP).

At Layer 5 in the AppleTalk architecture, the Zone Information Protocol (ZIP) provides localized broadcast traffic.

AppleTalk Features

- **Peer-to-peer based networking**
- **Client lookups for services propagate in logical zones**
- **Addresses use 24 bits (Network.Node)**
- **Nodes dynamically acquire addresses**
- **Routing protocol is RTMP**
  - **Updates sent at 10-second intervals**
  - **Metric is hop count**
  - **Distance vector routing**

5

Clients use broadcasts to learn about available services. The AppleTalk environment allows propagation of lookups by the router, ensuring that all available services will be located by the user.

AppleTalk addresses are composed of a 16-bit network number and an 8-bit node number.

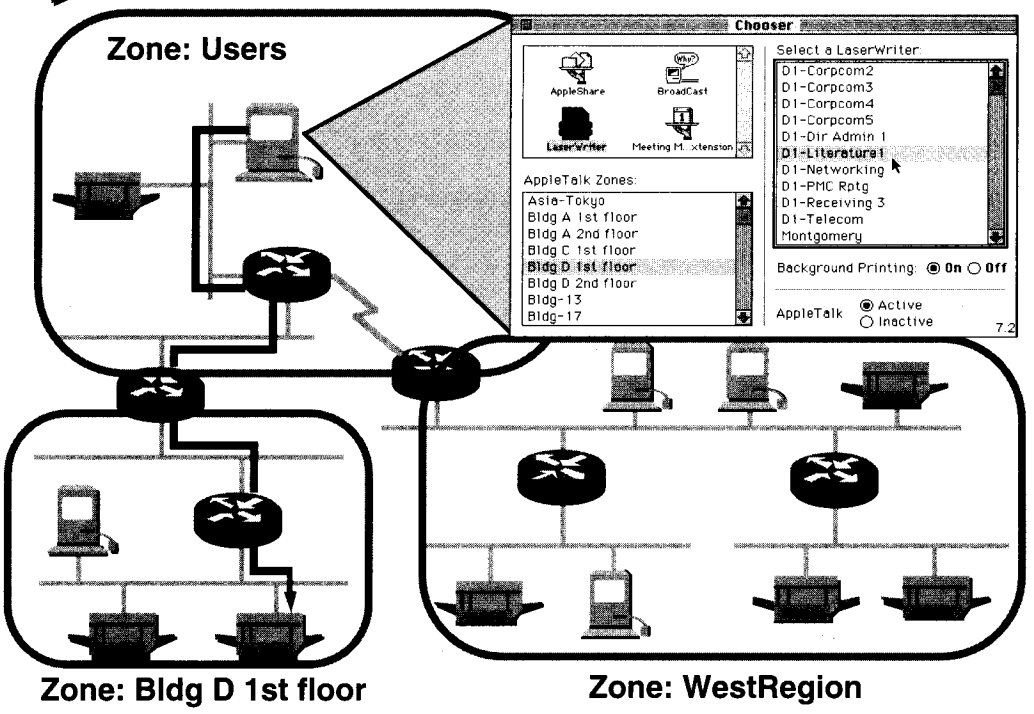
The network portion of the address is manually configured by the administrator. The node identifier portion is dynamically acquired during router startup.

The node identifier is also manually configured on the Cisco router. This process is useful when configuring AppleTalk for multipoint WANs and for dialers that use maps for WAN access.

RTMP provides routing information updates at Layer 4. RTMP is a Routing Information Protocol (RIP) derivative, using hop count as its metric for routing decisions.

Hosts listen to RTMP updates to learn the router's address.

AppleTalk Services



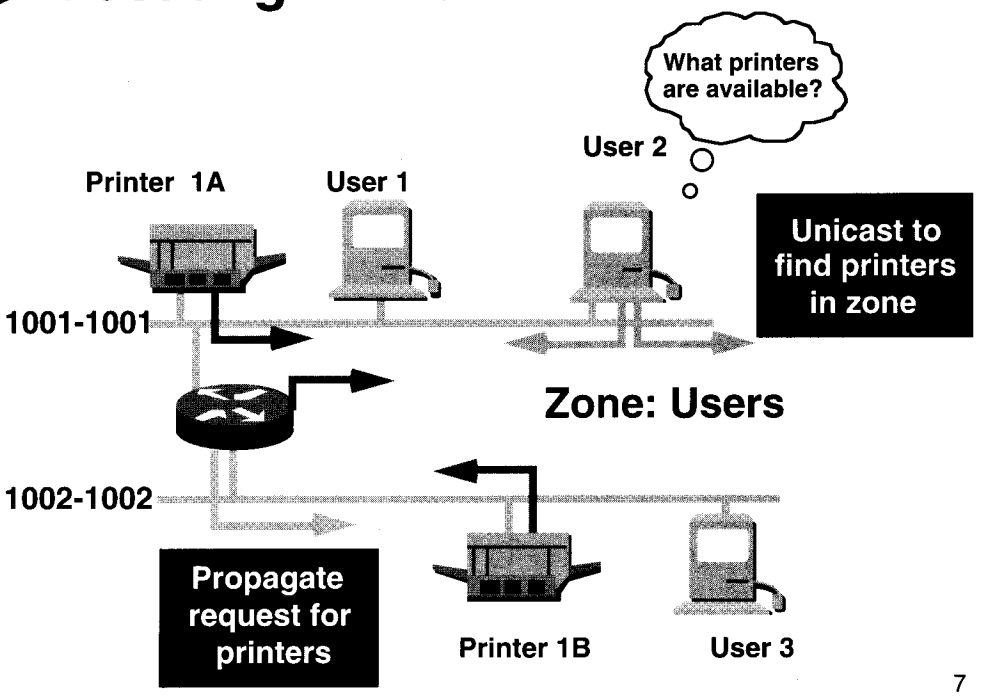
Nodes are assembled in logical groups called zones. Each device can be in only one zone.

When a Macintosh user requires a service:

- The Chooser sends a request to the router for a list of zones.
- The NBP looks for the servers in the zone that the Macintosh user specifies.
- The router forwards the request to each cable grouped in the selected zone.
- A multicast (one-to-many) goes to all devices that match the device type requested.
- Available matching services reply to the address of the Macintosh that originated the NBP process.
- Routers in the path forward these replies until they reach the originating router.
- The originating router sends the reply to the end user. The user selects the preferred service.

A logical link for that service is retained in the Macintosh for future reference, and a list of services and zones is maintained within the router for local reference.

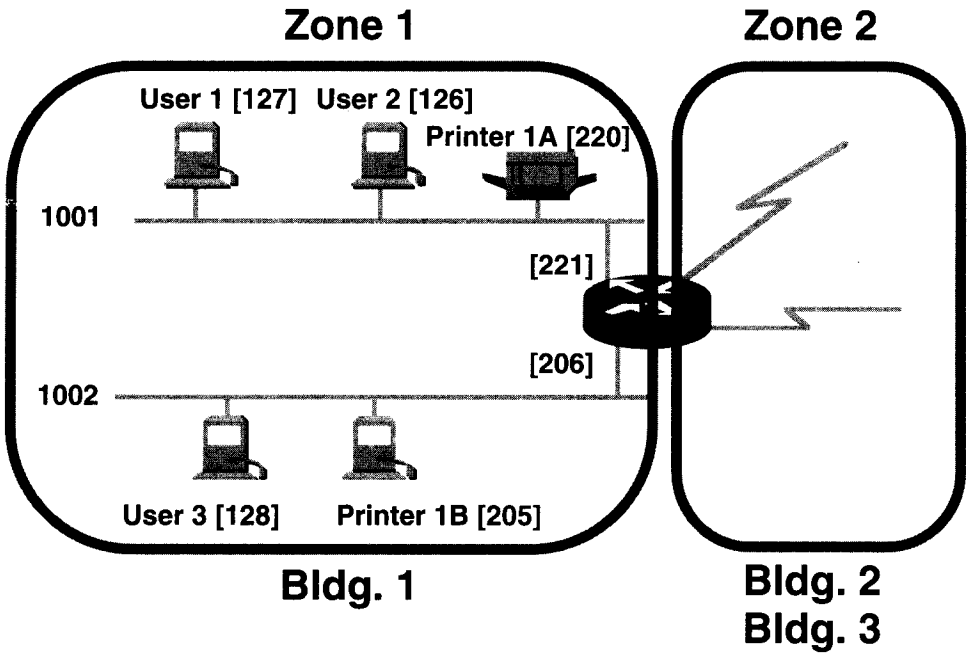
Locating Printers



Users on the AppleTalk network locate specific services using NBP requests.

In the graphic, user 2 looks for printers in the zone named Users. The router will create one request to send out cable 1001-1001, and another request to send out cable 1002-1002. Responses that the router forwards to user 2 informs user 2 about printer 1A and printer 1B.

Limiting Requests for Services



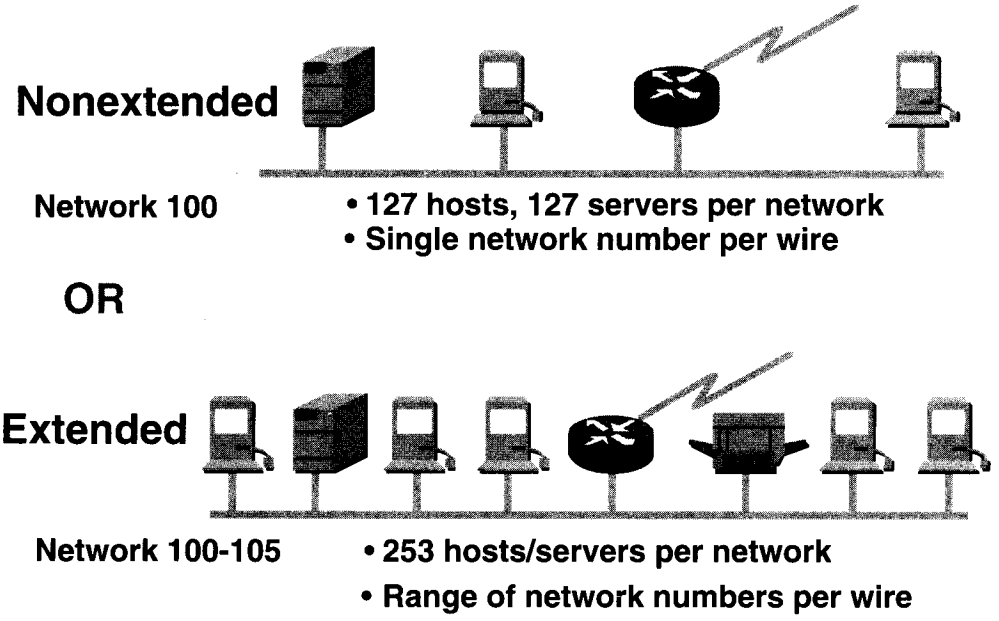
8

One method for controlling broadcast traffic is to allocate nodes to zones. A node can be in only one zone.

Each interface in the router must be assigned to a zone as part of its configuration.

Many devices, including the Cisco router interfaces, are visible in the default zone for a cable range.

Nonextended or Extended Networks

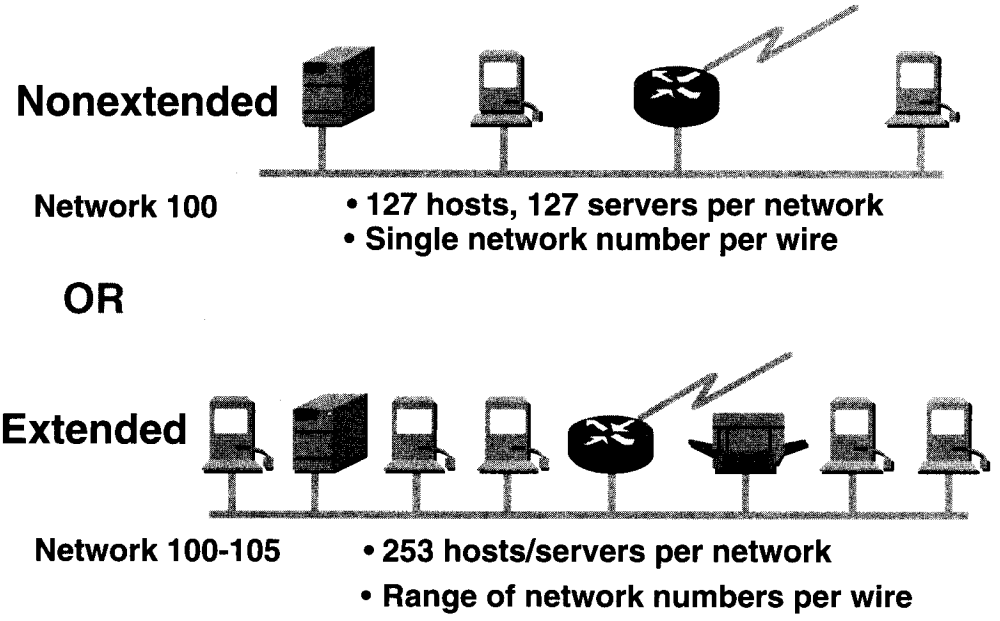


9

Early releases of AppleTalk (pre-1988) used a scheme referred to as Phase 1. This scheme did not allow large numbers of hosts on a single wire. An equal number of servers and hosts was allocated. Any Macintosh can be a host or server. The network and node address were considered separately, limiting the available address space.

Later releases of AppleTalk use extended addressing. Multiple network numbers can exist on the same wire. The maximum number of servers and hosts is the same as before 1988. The network and node addresses are considered in combination, greatly enlarging the available address space.

Nonextended or Extended Networks

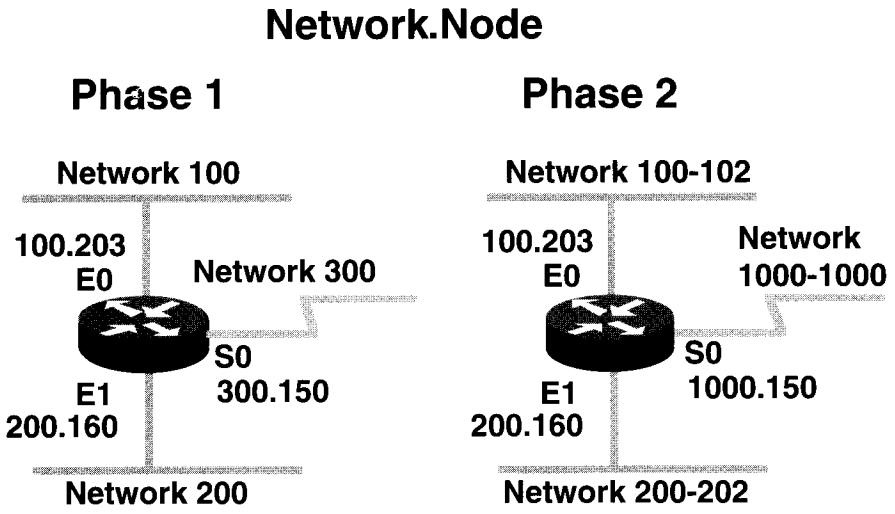


9

Early releases of AppleTalk (pre-1988) used a scheme referred to as Phase 1. This scheme did not allow large numbers of hosts on a single wire. An equal number of servers and hosts was allocated. Any Macintosh can be a host or server. The network and node address were considered separately, limiting the available address space.

Later releases of AppleTalk use extended addressing. Multiple network numbers can exist on the same wire. The maximum number of servers and hosts is the same as before 1988. The network and node addresses are considered in combination, greatly enlarging the available address space.

AppleTalk Addressing



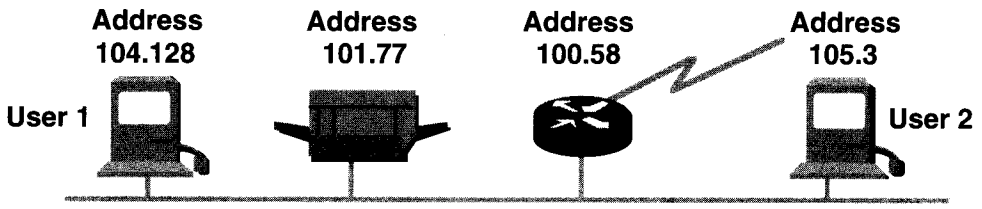
- **Interfaces use unique network numbers (ranges)**

10

In a Phase 1 network, only one network number is allowed for each wire. Node numbers are assigned dynamically when clients start up. In a Phase 2 network, multiple network numbers are available for each wire. The range of numbers is assigned by the administrator. Node numbers from 1 to 253 are assigned dynamically for both hosts and routers.

Both addressing schemes require that a unique network.node address be applied to each router interface.

Extended Addressing



- **Range of network numbers per wire**

11

In an extended network, the network numbers of the nodes can be different. There may be a wide network range on a single logical network.

Network number—16 bits

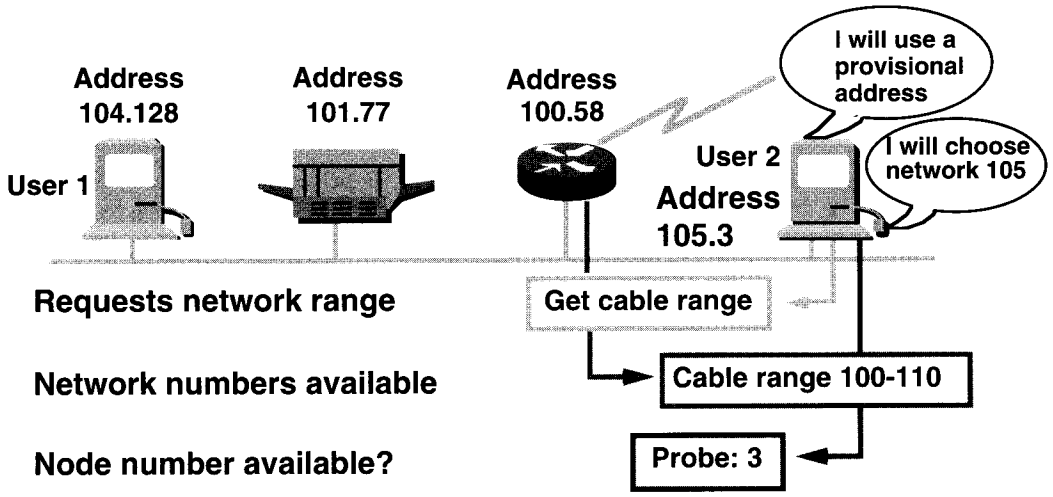
- A cable range states the span of network numbers available on this media.
- Narrow (unary) range networks are supported.
- A network number of 0 is reserved by the protocol for a newly attached node to use when it does not yet know the network number to use on its attached cable.

Node number—8 bits

- Numbers in the range 1 to 253 represent any node (user, printer, and other devices).
- The numbers 0, 254, and 255 are reserved on extended networks.

Node numbers are dynamically assigned.

AppleTalk Address Acquisition



12

User 2 is powered on, but has no address stored in its permanent memory (RAM). User 2's software selects a provisional network address from the FF00-FFE0 range and a random node number. The new node sends 10 AppleTalk Address Resolution Protocol (AARP) probes to verify the node ID availability.

A "get cable range" ZIP request is issued by user 2.

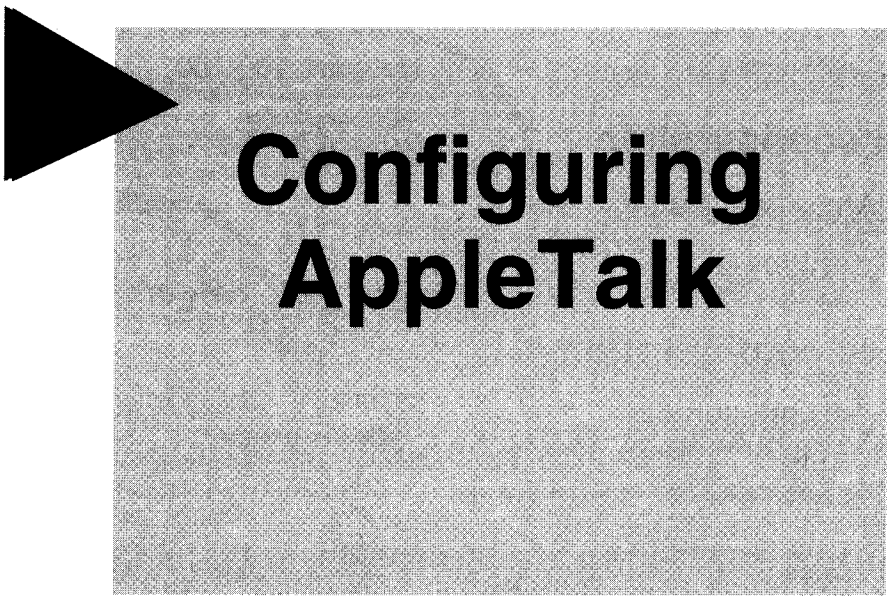
The router's response indicates the range of network numbers available on the wire. User 2 selects a network number from the cable range.

User 2 issues AARP probes for a node ID:

- If there is a response that the node ID is in use, user 2 tries another node ID.
- If there is no response to the probe, user 2 uses this ID.

User 2's address becomes 105.3.

After an address is acquired, it is saved in RAM. The stored address is probed for at the next power-up sequence, and if it is in use, dynamic assignment is initiated.

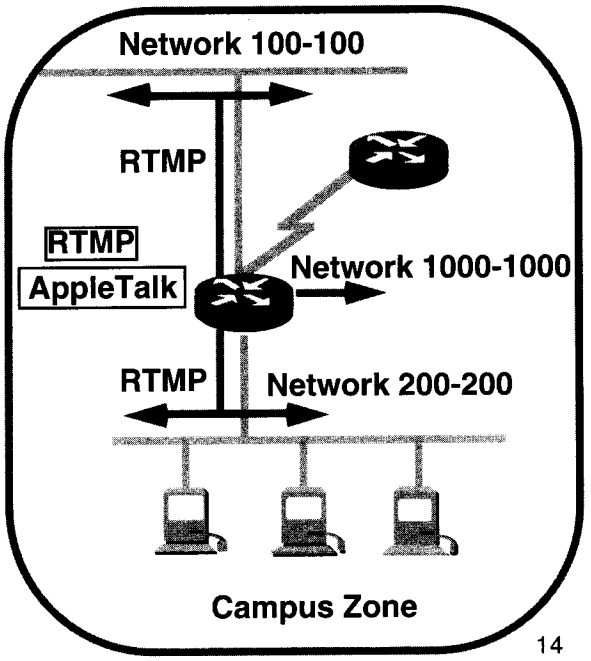


13

Configuring AppleTalk

AppleTalk Configuration Tasks

- **Global configuration**
  - **Select AppleTalk routing**
- **Interface configuration**
  - **Assign network number range** CABLE RANGE
  - **Select routing update protocol**
  - **Assign zones**



14

Configuration of AppleTalk as a routing protocol requires setting both global and interface parameters.

Global task: Select AppleTalk routing to start the routing process.

Interface tasks:

- Assign a range of network numbers to each interface. A narrow range can be an appropriate assignment.
- Assign each interface to a zone. Phase 2 allows multiple zones per segment.

After an address and zone name are assigned, the interface is enabled for packet processing. All routers in a network or data link must agree on the cable range, default zone, and zone list.

AppleTalk Configuration

Router (config) #

appletalk routing

- **Turns on AppleTalk routing**

Router (config-if) #

appletalk protocol { rtmp | eigrp | aurp }

- **Selects the protocol that generates routing updates on this interface**

15

The **appletalk routing** command starts the AppleTalk routing process.

The **appletalk protocol** command selects one or more routing protocols for use on this interface.

appletalk protocol Command	Description
rtmp	Routing protocol is RTMP, which is the default.
eigrp	Specifies that the routing protocol to use is Enhanced IGRP.
aurp	Specifies that the routing protocol to use is AURP. You can enable AURP only on tunnel interfaces.

If the **appletalk protocol** command is omitted in the interface specification, RTMP is selected by default.

AppleTalk Configuration (cont.)

Router (config-if) #

appletalk cable-range *cable-range* [*network.node*]

- **Assigns a range of network numbers**

Router (config-if) #

appletalk zone *zone-name*

- **Defines zone name**

16

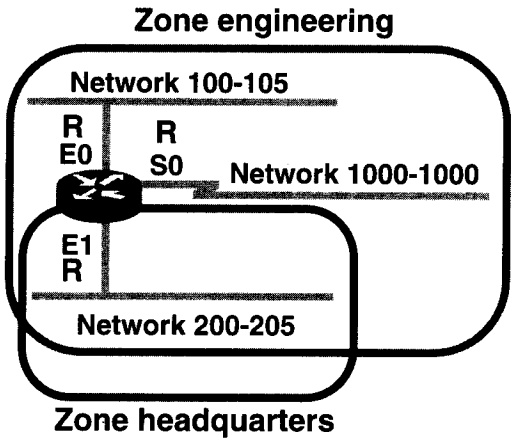
The **appletalk cable-range** command specifies a range of network numbers available to the interface. If the cable range value is 0-0, the interface is placed in discovery mode.

The optional *network.node* argument allows the network administrator to specify a unique address. This is useful on mapped interfaces.

The **appletalk zone** command assigns the zone name to the data link. Multiple zones can be assigned to one interface in a Phase 2 installation. The first zone name is the default zone name.

AppleTalk Configuration Example

```
appletalk routing

interface ethernet 0
appletalk cable-range 100-105
appletalk zone engineering

interface ethernet 1
appletalk cable-range 200-205
appletalk zone engineering
appletalk zone headquarters

interface serial 0
appletalk cable-range 1000-1000 1000.128
appletalk zone engineering
```



R = AppleTalk RTMP

17

In the example:

Command	Description
appletalk routing	Starts the AppleTalk routing process.
appletalk cable-range 100-105	Establishes a range of six network numbers available to devices on E0.
appletalk zone engineering	Places interface E0 into a zone named engineering.
appletalk cable-range 1000-1000 1000.128	Assigns a narrow cable range of 1000 to interface serial 0 and specifies the network.node address of 1000.128.

All interfaces are using RTMP as the default routing protocol because no **appletalk protocol** commands are specified.

After AppleTalk routing is enabled, interface E0 dynamically acquires a node number on one of six available network numbers. Serial 0 has a hard-coded address of 100.128. All interfaces in the router are part of the zone engineering, and E1 is also part of zone headquarters.

Discovery Mode

Phase 2

Router (config-if) #

appletalk cable-range 0-0

or

Router (config-if) #

appletalk cable-range *cable-range*
appletalk discovery

- **Enables interface to learn cable range and zone name**

18

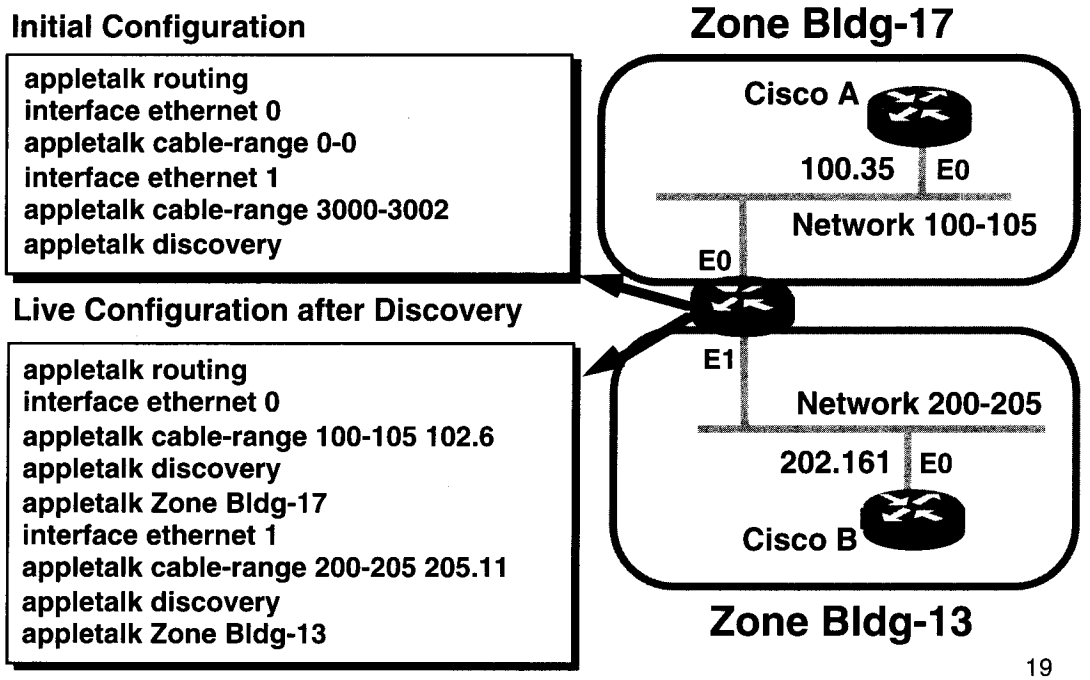
Discovery mode can occur if the router is not a seed router. Seed routers seed the AppleTalk internet with configuration information (like network number ranges and zones). The network administrator sets up a router as a seed router so the seed router provides its configuration information to other nonseed routers.

Placing the nonseed router interface in discovery mode allows the interface to dynamically learn its cable range and zone information from a seed router. There are two ways to place an interface into discovery mode:

- Phase 2—Assign the cable range as 0-0.
- Assign an address to the interface using normal configuration steps and then allow dynamic learning from other routers by using the default **appletalk discovery**. Do not use AppleTalk discovery on serial interfaces.

Note The second method shown (**appletalk discovery**) is on by default. To disable it, use the command **no appletalk discovery**.

Discovery Mode Example



In the example:

Command	Description
appletalk cable-range 0-0	Places E0 into discovery mode.
appletalk cable-range 3000-3002	Assigns a network range to E1.
appletalk discovery	Places E1 into discovery mode.

Both E0 and E1 dynamically learn their addresses and zones.

In the live configuration file, for E0:

Command	Description
appletalk cable-range 100-105	Is the acquired network range.
appletalk Zone Bldg-17	Is the acquired zone name.
appletalk discovery	Is the statement inserted when interface is placed in discovery mode. Remove this statement if dynamic or hard-coded addresses are used.

In the live configuration file, for E1:

Command	Description
appletalk cable-range 200-205	Is the acquired network range.
appletalk Zone Bldg-13	Is the acquired zone name.

Monitoring AppleTalk

```
Router# show appletalk interface ethernet 0
Ethernet0 is up, line protocol is up
  AppleTalk cable range is 3010-3019
  AppleTalk address is 3012.93, Valid
  AppleTalk zone is "ld-e0"
  AppleTalk port configuration verified by 3017.170
  AppleTalk address gleaning is enabled
  AppleTalk route cache is enabled
```

20

Use the **show appletalk interface** command to display status about all AppleTalk interfaces, including individual addressing, line status, timers, access lists assigned, and other details.

This command is particularly useful when you first enable AppleTalk on a router interface.

This display shows you this information:

- The interface is Ethernet 0.
- The cable range contains an address value from which an address was selected. The address is marked as valid.
- The zone name is listed.
- AppleTalk address gleaning is enabled.
- AppleTalk route cache is enabled.

Monitoring AppleTalk (cont.)

```
Router# show appletalk route
Codes: R - RTMP derived, E - EIGRP derived, C - connected, A - AURP
       S - static  P - proxy
5 routes in internet

The first zone listed for each entry is its default (primary) zone.

C Net 3000-3005 directly connected, Ethernet1, zone ozone
C Net 3010-3019 directly connected, Ethernet0, zone ld-e0
C Net 3020-3020 directly connected, Serial0, zone dc-s0
C Net 3021-3021 directly connected, Serial1, zone dc-s1
R Net 3030-3039 [1/G] via 3020.25, 4 sec, Serial0, zone cf-e0
```

G- Good
B- Bad
S- Suspect

Use the **show appletalk route** command to display the contents of the AppleTalk routing table.

The sample shows the zones assigned to each cable range. The highlighted line shows an example of a wide cable range in the entry derived from RTMP.

Monitoring AppleTalk (cont.)

```
Router# show appletalk zone
Name                                    Network(s)
ld-e0                                   3010-3019  3000-3005
ozone                                   3000-3005
cf-e0                                   3030-3039
dc-s0                                   3020-3020
dc-s1                                   3021-3021
```

22

The **show appletalk zone** command displays entries in the AppleTalk zone information table.

Notice that the wide range of networks, 3000-3005, occur in zone ld-e0 as well as in zone ozone. The NBP lookup process is limited to the zone specified by the Macintosh users zone selection in the Chooser.

Monitoring AppleTalk (cont.)

```
Router# show appletalk globals
AppleTalk global information:
  Internet is incompatible with older, AT Phase1, routers.
  There are 5 routes in the internet.
  There are 5 zones defined.
  Logging of significant AppleTalk events is disabled.
  ZIP resends queries every 10 seconds.
  RTMP updates are sent every 10 seconds.
  RTMP entries are considered BAD after 20 seconds.
  RTMP entries are discarded after 60 seconds.
  AARP probe retransmit count: 10, interval: 200
  AARP request retransmit count: 5, interval: 1000.
  DDP datagrams will be checksummed.
  RTMP datagrams will be strictly checked.
  RTMP routes may not be propagated without zones.
  Routes will not be distributed between routing protocols
  AppleTalk EIGRP is not enabled
  IPTalk uses the udp base port of 768 (Default).
  Alternate node address format will not be displayed.
  Access control of any networks of a zone hides the zone.
```


The **show appletalk globals** command displays information and settings about the router's global AppleTalk configuration parameters. The highlighted line indicates Phase 1 compatibility through the use of unary cable ranges and single zones per interface.

Monitoring AppleTalk (cont.)

```
Router# debug apple routing
AppleTalk RTMP routing debugging is on
AppleTalk EIGRP routing debugging is on
Router#
AT: RTMP from 3002.5 (new 0,old 0,bad 0,ign 0, dwn 0)
AT: RTMP from 3017.170 (new 0,old 0,bad 0,ign 0, dwn 0)
AT: src=Ethernet0:3012.93, dst=3010-3019, size=34, 4 rtes, RTMP pkt sent
AT: src=Ethernet1:3000.175, dst=3000-3005, size=34, 4 rtes, RTMP pkt sent
AT: src=Serial0:3020.26, dst=3020-3020, size=28, 3 rtes, RTMP pkt sent
AT: src=Serial1:3021.144, dst=3021-3021, size=34, 4 rtes, RTMP pkt sent
AT: Route ager starting on Main AT RoutingTable (5 active nodes)
AT: Route ager finished on Main AT RoutingTable (5 active nodes)
AT: RTMP from 3020.25 (new 0,old 1,bad 0,ign 1, dwn 0)
AT: RTMP from 3021.193 (new 0,old 1,bad 0,ign 3, dwn 0)
AT: RTMP from 3020.25 (new 0,old 1,bad 0,ign 1, dwn 0)
AT: RTMP from 3002.5 (new 0,old 0,bad 0,ign 0, dwn 0)
AT: RTMP from 3017.170 (new 0,old 0,bad 0,ign 0, dwn 0)
AT: src=Ethernet0:3012.93, dst=3010-3019, size=34, 4 rtes, RTMP pkt sent
AT: src=Ethernet1:3000.175, dst=3000-3005, size=34, 4 rtes, RTMP pkt sent
AT: src=Serial0:3020.26, dst=3020-3020, size=28, 3 rtes, RTMP pkt sent
AT: src=Serial1:3021.144, dst=3021-3021, size=34, 4 rtes, RTMP pkt sent
```

24

The **debug apple routing** command displays output from the RTMP routines. This command is used to monitor acquisition, aging, and advertisement of routes. It also reports conflicting network numbers on the same network.

Summary

(DECIMAL)

AppleTalk addressing is Network.Node

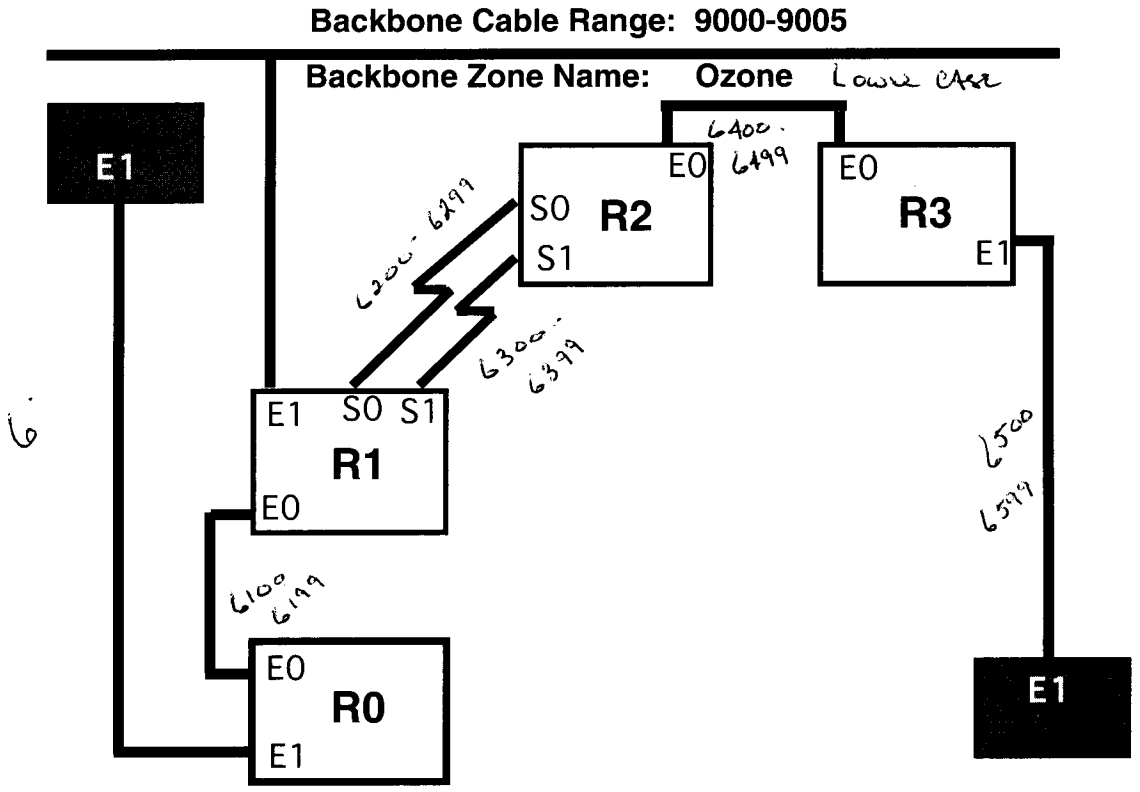
Addresses are dynamically acquired

Multiple network numbers can exist on one wire

Unicast traffic is limited using zones

25

Lab: AppleTalk Planning and Implementation



Objective: Enable AppleTalk protocol and configure AppleTalk interfaces.

Objective: Monitor AppleTalk operation in the router.

Instructions: Enable the AppleTalk protocol and configure the interfaces, then monitor AppleTalk operation in the router. Use the assigned AppleTalk cable ranges for your own workgroup as shown. For the interface to the backbone use the cable range 9000-9005 and the zone name Ozone.

Workgroup	Address Range
A	1000-1999
B	2000-2999
C	3000-3999
D	4000-4999
E	5000-5999
F	6000-6999
Backbone	9000-9005

AppleTalk Planning Worksheet

Step 1 Given the address range, assign an AppleTalk cable range for each link in your workgroup.

Step 2 Agree on a unique zone name for each link between your router and adjacent routers. fzone

Step 3 Assign a zone name for your entire group. fzone

Step 4 Use only the cable range and zone name provided on the previous page for the backbone connection (E1 on R1).

Step 5 Negotiate with the neighbor workgroups connected to E1 on R3 and E1 on R0 to determine the cable range and zone names for interfaces on the shared links.

f

Group: fzone Address Range: 6000 - 6999 Workgroup Zone Name: fzone

Router R0 host name:

Interface	Cable Range	Zone Name
E0		
E1		

Router R1 host name:

Interface	Cable Range	Zone Name
E0	6100 - 6199	zone 61
E1	~~6200~~ - 9000 - 9005	ozone
S0	6200 - 6299	zone 62
S1	6300 - 6299	zone 63

Router R2 host name:

Interface	Cable Range	Zone Name
E0		
S0		
S1		

Router R3 host name:

Interface	Cable Range	Zone Name
E0		
E1		

AppleTalk Implementation

Instructions: Using the work from the previous lab, this lab will form an AppleTalk internetwork within the training lab by configuring AppleTalk, establishing connectivity within each workgroup, and expanding the network connectivity to incorporate the rest of the groups in the room.

Step 1 Shut down all E1 interfaces that connect your network to other groups.

Step 2 Using the **configure** command, start Appletalk routing.

Step 3 Assign the AppleTalk cable range and appropriate zone names to the interfaces.

Step 4 Use the **show appletalk interface** command to verify address assignment.

Step 5 Use the **show appletalk route** command to verify entries in the routing table.

Step 6 Use the **ping** command to verify connectivity across your workgroup.

Step 7 Reactivate E1 interfaces and expand your network to incorporate the room.

Step 8 Use the IP addresses of the E0 interface to **telnet** to routers in other groups in the network.

Step 9 Use the **show appletalk interface** command to discover the AppleTalk addresses on remote routers.

Step 10 Use the **ping** command to verify Apple connectivity across all workgroups.

Step 11 Once you have confirmed that connectivity has been established, save the configuration of your router in nonvolatile memory.

CHAPTER 15

Basic Traffic Management with Access Lists

Objectives

Upon completion of this chapter, you will be able to perform the following tasks:

Describe the use, value, and processes of access lists

Configure standard and extended access lists to filter IP traffic

Configure IPX access lists and SAP filters to control basic Novell traffic

Configure cable-range access lists and zone filters to control basic AppleTalk traffic

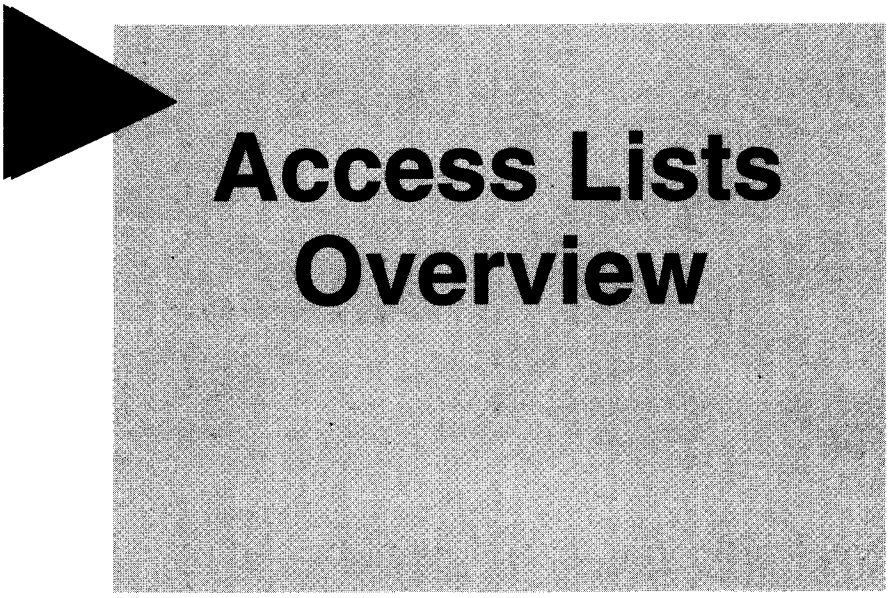
Monitor and verify selected access list operations on the router

2

This chapter presents basic and extended access lists as a means to control network traffic. It explains general concepts about access lists. Following the general explanation, it explains how to configure IP, IPX, and AppleTalk access lists. There is a lab included for each protocol.

Sections:

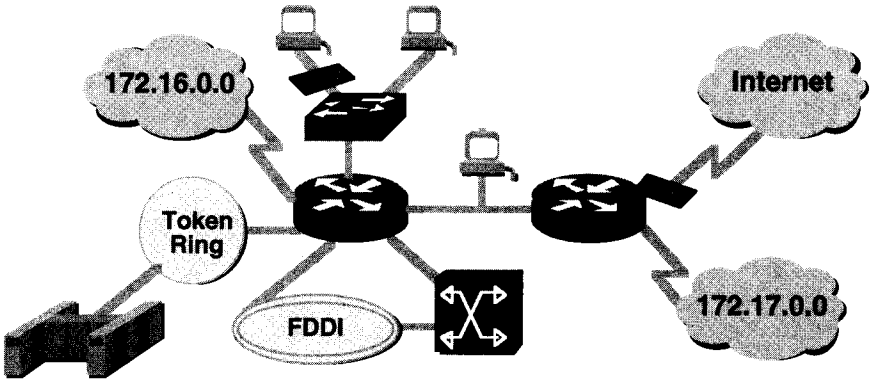
- Access Lists Overview
- TCP/IP Access Lists
- Novell IPX Access Lists
- AppleTalk Access Lists
- Answers to Exercise



3

Access Lists Overview

Why Use Access Lists?



- **Deny traffic you do not want based on packet tests (for example, addressing or traffic type)**

4

The earliest routed networks connected a modest scale of LANs and hosts. Next, the network administrator enlarged router connections to legacy and outside partners' networks. Increased use of the Internet brought new challenges to access control. Newer technology—from optical backbones to broadband services and high-speed LAN switches—increased control challenges again.

Network administrators face the following dilemma: how to deny unwanted connections while allowing appropriate access? Although other tools such as passwords, callback equipment, and physical security devices are helpful, they often lack the flexible expression and specific controls most administrators prefer.

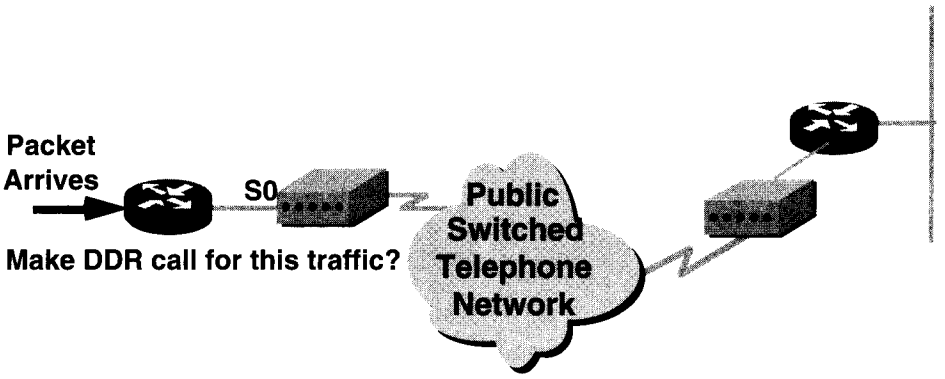
Access lists offer another powerful tool for network control. These lists add the flexibility to filter the packet flow that flow in or out router interfaces. The access lists help protect expanding network resources without impeding the flow of legitimate communication. Access lists differentiate packet traffic into categories that permit or deny other features. You can also use access lists to:

- Identify packets for priority or custom queuing
- Restrict or reduce the contents of routing updates

Access lists also process packets for other security features to:

- Provide IP traffic dynamic access control with enhanced user authentication using the lock-and-key feature
- Identify packets for encryption
- Identify Telnet access allowed to the router virtual terminals

Why Use Access Lists? (cont.)



- **Specify packet traffic for dialing remote sites using dial-on-demand routing (DDR)**

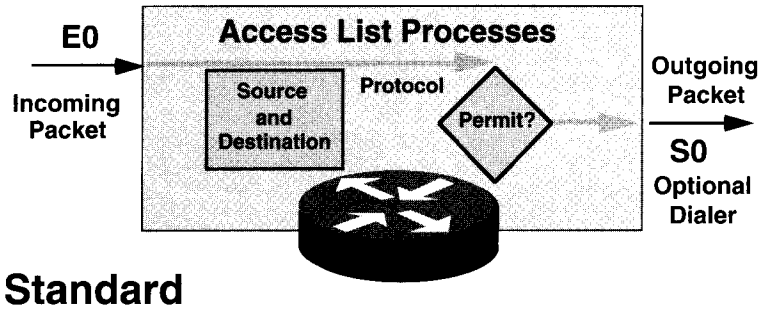
5

Compared to LAN or campus-based networking, the traffic that uses dial-on-demand routing (DDR) is typically low volume and periodic. DDR initiates a WAN call to a remote site only when there is traffic to transmit. To identify this traffic, you specify the packets that the DDR processes on the router will interpret as "interesting" traffic.

When you configure for DDR, you must enter configuration commands that indicate what protocol packets constitute interesting traffic to initiate the call. To do this, you will enter access-list statements to identify the source and destination addresses and choose specific protocol selection criteria for initiating the call.

Then you must establish the interfaces where the DDR call initiates. This step designates a dialer group. The dialer group associates the results of the access list specification of interesting packets to the router's interfaces for dialing a WAN call.

What Are Access Lists?



- **Standard**
  - **Simpler address specifications**
  - **Generally permits or denies entire protocol suite**
- **Extended**
  - **More complex address specifications**
  - **Generally permits or denies specific protocols**

6

Access lists are statements that specify conditions that an administrator sets so the router will handle the traffic covered by the access list in an out-of-the ordinary manner. Access lists give added control for processing the specific packets in a unique way. The two main types of access lists are:

- Standard access lists

  Standard access lists for IP check the source address of packets that could be routed. The result permits or denies output for an entire protocol suite, based on the network/subnet/host address.

  For example, packets coming in E0 are checked for address and protocol. If permitted, the packets are output through S0, which is grouped to the access list.

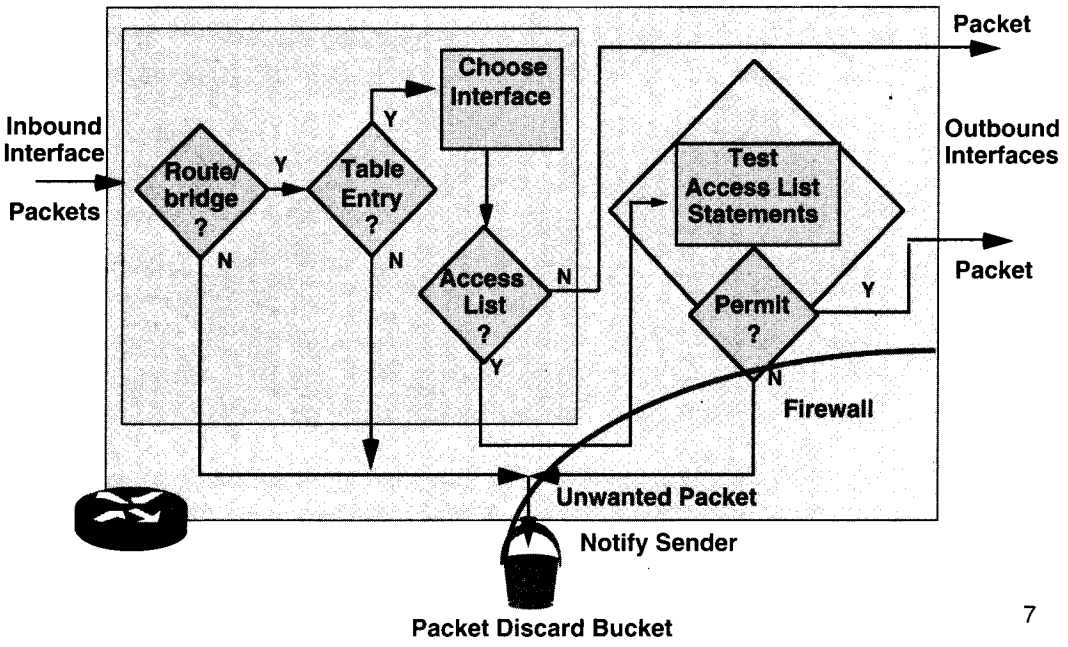
  If the packets are denied by the standard access list, all these packets for the given category are dropped.

- Extended access lists

  Extended access lists check for both source and destination packet addresses. They also can check for specific protocols, port numbers, and other parameters. This allows administrators more flexibility to describe what checking the access list will do. Packets can be permitted or denied output based on where the packet originated and on its destination.

  The extended access list also permits or denies with more granularity. For example, it can allow electronic mail traffic from E0 to specific S0 destinations, while denying remote logins or file transfers.

How Access Lists Work



Access lists express the set of rules that give added control for packets that enter inbound interfaces, packets that relay through the router, and packets that exit outbound interfaces of the router. Access lists do not act on packets that originates in the router itself.

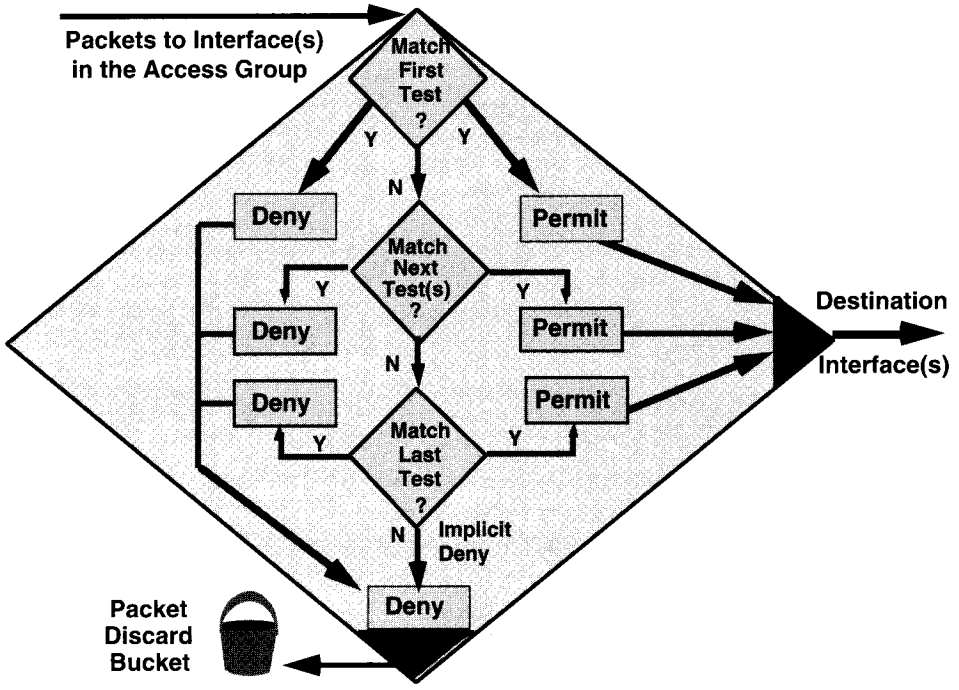
The beginning of the process is the same regardless of whether access lists are used: As a packet enters an interface, the router checks to see whether it is routable (or bridgeable). If either situation is false, the packet will be dropped. A routing table entry indicates a destination network, some routing metric or state, and the interface to use.

Next the router checks to see whether the destination interface is grouped to an access list. If it is not, the packet can be sent to the output buffer; for example, if it will use T0, which has no access lists in effect, the packet uses T0 directly.

Interface E0 has been grouped to an extended access list. The administrator used precise, logical expressions to set the access list. Before a packet can proceed to that interface, it is tested by a combination of access list statements associated with that interface.

Based on the extended access list tests, the packet can be permitted. For inbound lists, this means continue to process the packet after receiving it on an inbound interface. For outbound lists, this means send it to the output buffer for E0; otherwise test results can deny permission. This means discard the packet. The router's access list provides firewall control to deny use of the E0 interface. When discarding packets, some protocols return a special packet to the sender. This notifies the sender of the unreachable destination.

A List of Tests: Deny or Permit



Access list statements operate in sequential, logical order. They evaluate packets from the top down. If a packet header and access list statement match, the packet skips the rest of the statements. If a condition match is true, the packet is permitted or denied. There can be only one access list per protocol per interface.

In the graphic, for instance, by matching the first test, a packet is denied access to destination interfaces. It will be discarded and dropped into the bit bucket. The packet is not exposed to any access list tests that follow.

Only if the packet does not match conditions of the first test will it drop to the next access list statement. Assume a different packet's parameters match the next test, a permit statement; the permitted packet proceeds to the destination interface.

Another packet does not match the conditions of the first or second test, but does match conditions of the next access list statement; again, a permit results.

Note For logical completeness, an access list must have conditions that test true for all packets using the access list. A final implied statement covers all packets for which conditions did not test true. This final test condition matches all other packets. It results in a deny. Instead of proceeding in or out an interface, all these remaining packets are dropped.

THE ORDER OF THE LIST MATTERS. Processes FROM THE TOP.

Access List Command Overview

Step 1: Set parameters for this access list test statement (which can be one of several statements)

Router (config) #

access-list *access-list-number* { permit | deny } { test conditions }

Step 2: Enable an interface to become part of the group that uses the specified access list

Router (config-if) #

{ protocol } access-group *access-list-number*

- **Access lists are numbered (for IP, numbered or named)**

9

In practice, access list commands can be lengthy character strings. Access lists can be complicated to enter or interpret. However, you can simplify understanding the general access list configuration commands by reducing the commands to two general elements.

Step 1 The access list process contains global statements:

- This global statement identifies the access list, usually an access list number. This number refers to the type of access list this will be. In Cisco IOS Release 11.2 or newer, access lists for IP may also use an access list name rather than a number.
- The permit or deny term in the global access list statement indicates how packets that meet the test conditions will be handled by Cisco IOS. Permit usually means the packet will be allowed to use one or more interfaces that you will specify later.
- The final term or terms specifies the test conditions used by this access list statement. The test can be as simple as checking for a single source address, but usually test conditions are extended to include several test conditions. Use several global access list statements with the same identifier to stack several test conditions into a logical sequence or list of tests.

Step 2 The access list process uses an interface command. All the access list statements identified by the access-list number associate with one or more interfaces. Any packets that pass the access list test conditions can be permitted to use any interface in the access group of interfaces.

How to Identify Access Lists

Access List Type		Number Range/Identifier
IP	Standard Extended	1-99 100-199 Named (Cisco IOS 11.2 and later)
IPX	Standard SAP filters	800-899 1000-1099
AppleTalk		600-699

- Number identifies the protocol and type
- Other number ranges for most protocols

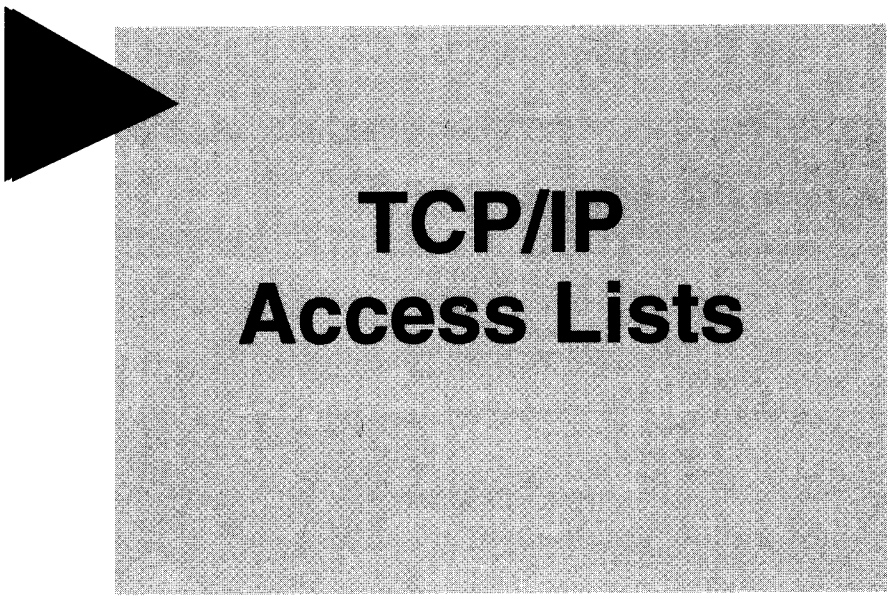
10

Access lists can control most protocols on a Cisco router. The graphic shows the protocols and number ranges of the access list types covered in this chapter.

An administrator enters a number in the protocol number range as the first argument of the global access list statement. The router identifies which access list software to use based on this numbered entry. Access list test conditions follow as arguments. These arguments specify tests according to the rules of the given protocol suite. The meaning or validity of the standard and extended identification scheme for access lists varies by protocol.

Many access lists are possible for a protocol. Select a different number from the protocol number range for each new access list; however, the administrator can specify only one access list per protocol per interface.

Note With Cisco IOS Release 11.2 and later you can also identify a standard or extended IP access list with an alphanumeric string (name) instead of the current numeric (1 to 199) representation. This can be an easier identification method to administer. Named IP access lists provide other advantages covered later in this chapter.

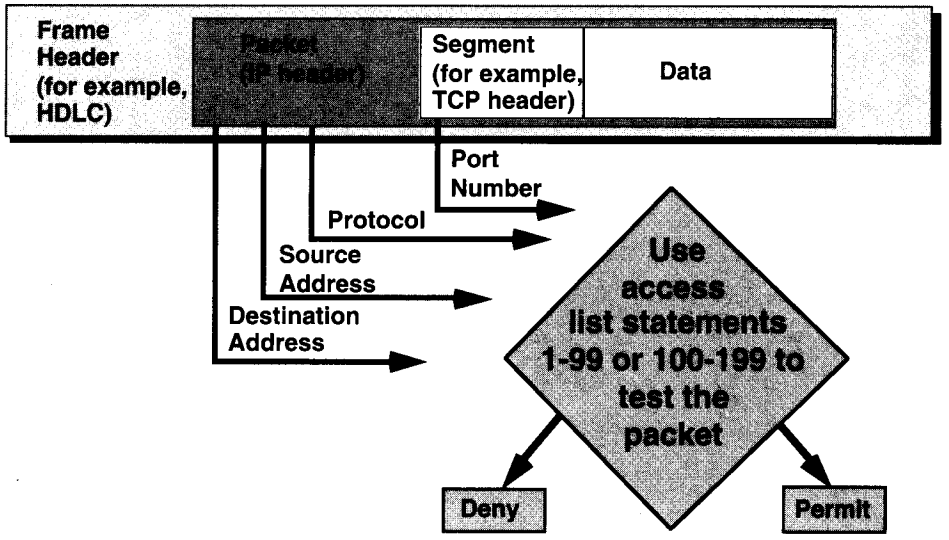


11

TCP/IP Access Lists

Testing Packets with Access Lists

An Example from a TCP/IP Packet



12

For TCP/IP packet filters, Cisco IOS access lists check the packet and upper-layer headers.

This course covers checking the packet for:

- Source IP addresses using standard access lists; identify these with a number in the range 1 to 99.
- Destination and source IP addresses or specific protocols using extended access lists; identify these with a number in the range 100 to 199.
- Upper-level TCP or UDP port numbers in addition to the other tests in extended access lists; also identify these with a number in the range 100 to 199.

For all of these TCP/IP access lists, after a packet is checked for a match with the access list statement, it can be denied or permitted to use an interface in the access group.

Key Concepts for IP Access Lists

- **Standard lists (1 to 99) test conditions of all IP packets from source addresses**
- **Extended lists (100 to 199) can test conditions of**
  - **Source and destination addresses**
  - **Specific TCP/IP-suite protocols**
  - **Destination ports**
- **Wildcard bits indicate how to check the corresponding address bits (0=check, 1=ignore)**

13

Create access lists using the normal global router configuration process.

Specifying an access list number from 1 to 99 instructs the router to accept standard IP access list statements. Specifying an access list number from 100 to 199 instructs the router to accept extended IP access list statements.

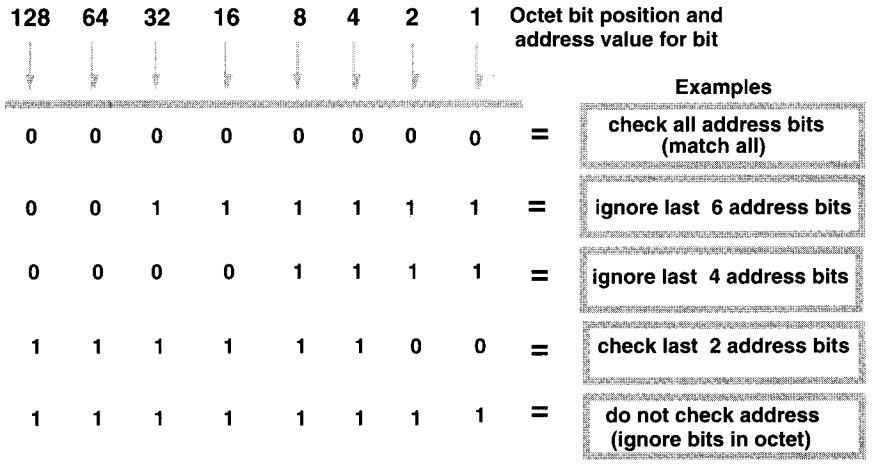
The administrator must carefully decide specific access controls logically and order the statements to achieve intended controls. Permitted protocols must be specified. All other TCP/IP protocols are denied.

Select which IP protocols to check. Any other IP protocols are not checked. Later in the procedure, the administrator can also specify an optional destination port for more granularity.

Address filtering occurs using access list address wildcard masking to identify how to check or ignore corresponding IP address bits.

MUST HAVE AT LEAST ONE "PERMIT" ACL

How to Use Wildcard Mask Bits



- **0 means check corresponding bit value**
- **1 means ignore value of corresponding bit**

14

IP access lists use wildcard masking. Wildcard masking for IP address bits uses the number 1 and the number 0 to identify how to treat the corresponding IP address bits.

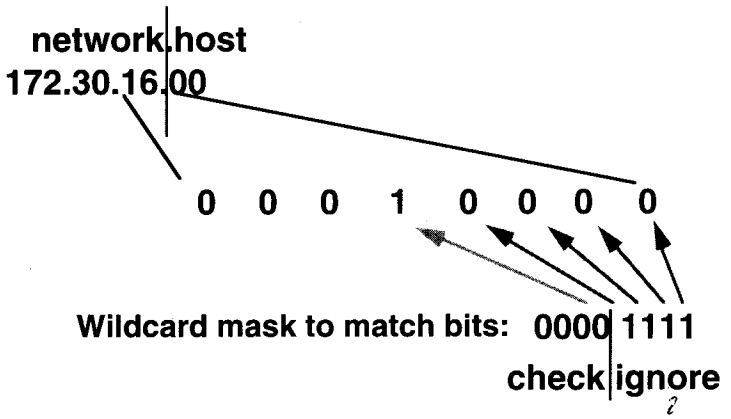
- A wildcard mask bit 0 means "check the corresponding bit value."
- A wildcard mask bit 1 means "do not check (ignore) that corresponding bit value."

By carefully setting wildcard masks, an administrator can select single or several IP addresses for permit or deny tests. Refer to the example in the graphic.

Note Wildcard masking for access lists operates differently from an IP subnet mask. A zero in a bit position of the access list mask indicates that the corresponding bit in the address must be checked; a one in a bit position of the access list mask indicates the corresponding bit in the address is not "interesting" and can be ignored.

How to Use Wildcard Mask Bits (cont.)

IP access list test conditions:
Check for IP subnets 172.30.16.0 to 172.30.31.0



- **Address and wildcard mask: 172.30.16.0 0.0.15.255**

15

You have seen how the zero and one bits in an access list wildcard mask cause the access list to either check or ignore the corresponding bit in the IP address. In the example, this wildcard masking process is applied in an example.

An administrator wants to test an IP address for subnets that will be permitted or denied. Assume the IP address is Class B (first two octets are the network number) with eight bits of subnetting (the third octet is for subnets). The administrator wants to use IP wildcard masking bits to match subnets 172.30.16.0 to 172.30.31.0. Here is how to use the wildcard mask to do this:

- To begin, the wildcard mask will check the first two octets (172.30) using corresponding zero bits in the wildcard mask.
- Because there is no interest in individual host addresses (a host ID will not be .00 at the end of the address), the wildcard mask will ignore the final octet using corresponding one bits in the wildcard mask.
- In the third octet, where the subnet address occurs, the wildcard mask will check that the bit position for the binary 16 is on and all the higher bits are off using corresponding zero bits in the wildcard mask.

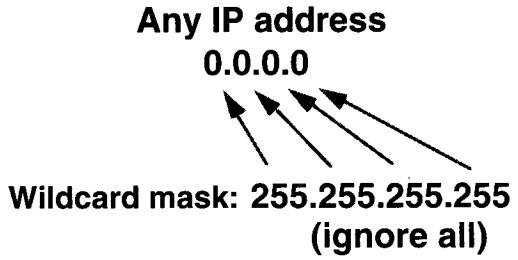
  For the final (low end) four bits in the this octet the wildcard mask will ignore the value—in these positions, the address value can be binary 0 or binary 1. In this way, the wildcard mask matches subnet 16, 17, 18, and so on up to subnet 31. The wildcard mask will not match any other subnets.

In this example, the address 172.30.16.0 with the wildcard mask 000.000.001.111 matches subnets 172.30.16.0 to 172.30.31.0.

16 00010000 16+15
15 00001111
00011111 = 31

How to Use the Wildcard *any*

Test conditions: Ignore all the address bits (match any)



- **Accept any address: 0.0.0.0 255.255.255.255; abbreviate the expression using the keyword *any***

16

Working with decimal representations of binary wildcard mask bits can be tedious.

For the most common uses of wildcard masking, you can use abbreviation words. These abbreviation words reduce how many numbers an administrator will be required to enter while configuring address test conditions. One example where you can use an abbreviation instead of a long wildcard mask string is when you want to match any address.

Consider a network administrator who wants to specify that any destination address will be permitted in an access list test. To indicate any IP address, the administrator would enter 0.0.0.0; then to indicate that the access list should ignore (allow without checking) any value, the corresponding wildcard mask bits for this address would be all ones (that is, 255.255.255.255).

The administrator can use the abbreviation *any* to communicate this same test condition to Cisco IOS access list software. Instead of typing 0.0.0.0 255.255.255.255, the administrator can use the word *any* by itself as the keyword.

How to Use the Wildcard *host*

Test conditions: Check all the address bits (match all)

An IP host address, for example:

172.30.16.29

Wildcard mask: 0.0.0.0

(check all bits)

- **Example 172.30.16.29 0.0.0.0 checks all the address bits**
- **Abbreviate the wildcard using the IP address followed by the keyword *host.* For example, 172.30.16.29 host**

17

A second common condition where Cisco IOS will permit an abbreviation term in the extended access list wildcard mask is when the administrator wants to match all the bits of an entire IP host address.

Consider a network administrator who wants to specify that a specific IP host address will be denied in an access list test. To indicate a host IP address, the administrator would enter the full address—for example, 172.30.16.29; then to indicate that the access list should check all the bits in the address, the corresponding wildcard mask bits for this address would be all zeros (that is, 0.0.0.0).

The administrator can use the abbreviation *host* to communicate this same test condition to Cisco IOS access list software. In the example, instead of typing 172.30.16.29 0.0.0.0, the administrator can use the word *host.* An example of using this abbreviation in as an access list test condition is the string *172.30.16.29 host.*

IP Standard Access Configuration

Router (config) #

access-list *access-list-number* { permit | deny }
***source* [*source-mask*]**

IF NO SOURCE MASK, ASSUMES 0's.

- **Sets parameters for this list entry**
- **IP standard access lists use 1 to 99**

Router (config-if) #

ip access-group *access-list-number* { in | out }

OUT IS DEFAULT SHOULD SPECIFY THOUGH

- **Activates the list on an interface**

18

The **access-list** command creates an entry in a standard traffic filter list.

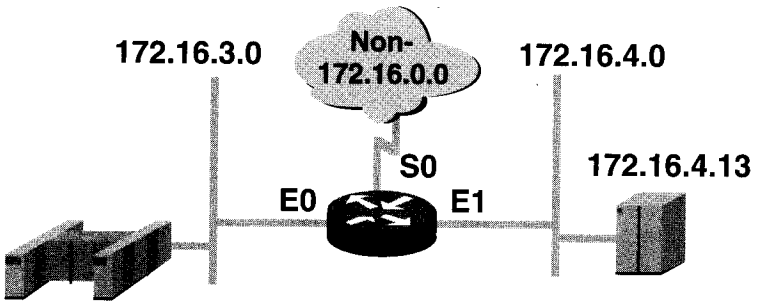
access-list Command	**Description**
access-list-number	Identifies the list to which the entry belongs; a number from 1 to 99.
permit | deny	Indicates whether this entry allows or blocks traffic from the specified address.
source	Identifies source IP address.
source-mask	Identifies which bits in the address field are matched. It has a 1 in positions indicating "don't care" bits, and a 0 in any position that is to be strictly followed.

The **ip access-group** command links an existing access list to an outbound interface. Only one access list per port per protocol per direction is allowed.

ip access-group Command	**Description**
access-list-number	Indicates the number of the access list to be linked to this interface.
in | out	Selects whether the access list is applied to the incoming or outgoing interface. If in or out is not specified, out is the default.

Note To remove an access list, first enter the **no access-group** command with all of its set parameters, then enter the **no access-list** command with all of its set parameters.

Standard Access List Example 1




```
access-list 1 permit 172.16.0.0  0.0.255.255
(implicit deny all - not visible in the list)
(access-list 1 deny 0.0.0.0   255.255.255.255)

interface ethernet 0
ip access-group 1 out
interface ethernet 1
ip access-group 1 out
```

- **Permit my network only**

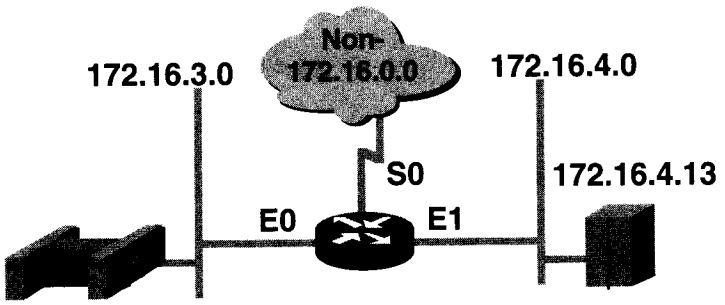
19

In the example:

access-list Command	Description
1	Access list number; indicates this is a simple list.
permit	Traffic that matches selected parameters will be forwarded.
172.16.0.0	IP address that will be used with the wildcard mask to identify the source network.
0.0.255.255	Wildcard mask; 0s indicate positions that must match, 1s indicate "don't care" positions.
ip access-group 1 out Command	Links the access list to an outgoing interface.

This access list allows only traffic from source network 172.16.0.0 to be forwarded. Non-172.16.0.0 network traffic is blocked.

Standard Access List Example 2




```
access-list 1 deny 172.16.4.13 host
access-list 1 permit 0.0.0.0 255.255.255.255
(implicit deny all)
(access-list 1 deny 0.0.0.0 255.255.255.255)

interface ethernet 0
ip access-group 1
```

IMPLICIT DENY WOULD NEVER BE EXECUTED, EVERY PACKET WOULD BE PROCESSED BY ONE OF THE FIRST TWO STATEMENTS.

- **Deny a specific host**

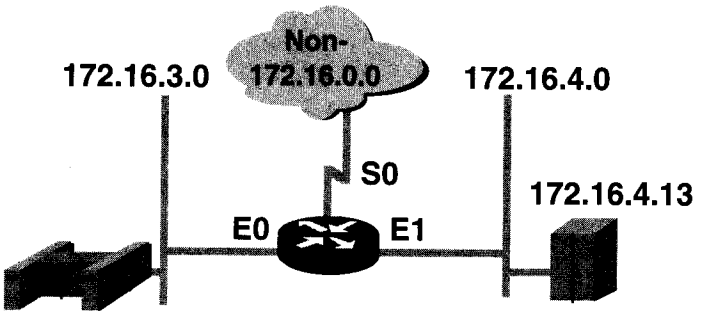
In the example:

access-list Command	Description
1	Access list number; indicates this is a simple list.
deny	Traffic that matches selected parameters will not be forwarded.
172.16.4.13	IP address of the source host.
0.0.0.0	Wildcard mask; 0s indicate positions that must match, 1s indicate "don't care" positions. All 0s in the mask indicates that all 32 bits will be checked in the source address.

access-list Command	Description
1	Access list number; indicates this is a simple list.
permit	Traffic that matches selected parameters will be forwarded
0.0.0.0	IP address of the source host; all 0s indicate a placeholder.
255.255.255.255	Wildcard mask; 0s indicate positions that must match, 1s indicate "don't care" positions. All 1s in the mask indicates that all 32 bits will not be checked in the source address.

This access list is designed to block traffic from a specific address, 172.16.4.13, and to allow all other traffic to be forwarded on interface Ethernet 0.

Standard Access List Example 3




```
access-list 1 deny 172.16.4.0  0.0.0.255
access-list 1 permit any
(implicit deny all)
(access-list 1 deny 0.0.0.0   255.255.255.255)

interface ethernet 0
ip access-group 1
```

- **Deny a specific subnet**

21

In the example:

access-list Command	Description
1	Access list number; indicates this is a simple list.
deny	Traffic that matches selected parameters will not be forwarded.
172.16.4.0	IP address of the source subnet.
0.0.0.255	Wildcard mask; 0s indicate positions that must match, 1s indicate "don't care" positions. The mask with 0s in the first three octets indicates those positions must match; the 255 in the last octet indicates a "don't care" condition.

access-list Command	Description
1	Access list number; indicates this is a simple list.
permit	Traffic that matches selected parameters will be forwarded.
any	Abbreviation for the IP address of the source; all 0s indicate a placeholder and the wildcard mask 255.255.255.255. All 1s in the mask indicates that all 32 bits will not be checked in the source address.

This access list is designed to block traffic from a specific subnet, 172.16.4.0, and to allow all other traffic to be forwarded.

Extended IP Access Lists

- **Allow more precise filtering conditions**
  - **— Check source and destination IP address**
  - **— Specify an optional IP protocol port number**
  - **— Use access list number range 100 to 199**

22

The standard access list (numbered 1 to 99) may not provide the traffic-filtering control you need. Standard access lists filter based on a source address and mask. Standard access lists permit or deny the entire TCP/IP protocol suite. You may need a more precise way to configure your firewall policy.

For more precise traffic-filtering control, use extended IP access lists. Extended IP access list statements check for source address and for destination address. In addition, at the end of the extended access list statement, you gain additional precision from a field that specifies the optional TCP or UDP protocol port number. These can be the well-known port numbers for TCP/IP. A few of the most common port numbers are as follows:

Well-Known Port Number (Decimal)	IP Protocol
20	File Transfer Protocol (FTP) data
21	FTP program
23	Telnet
25	Simple Mail Transport Protocol (SMTP)
69	Trivial File Transfer Protocol (TFTP)
53	Domain Name System (DNS)

By using this option, you can specify the logical operation the extended access list will perform on specific protocols. Extended access lists use a number from the range 100 to 199.

Extended Access List Configuration

Router (config) #

access-list *access-list-number* { permit | deny } *protocol source source-mask destination destination-mask* [*operator operand*] [established]

- **Sets parameters for this list entry**
- **IP uses a list number in range 100 to 199**

ip access-group *access-list-number* { in | out }

- **Activates the extended list on an interface**

23

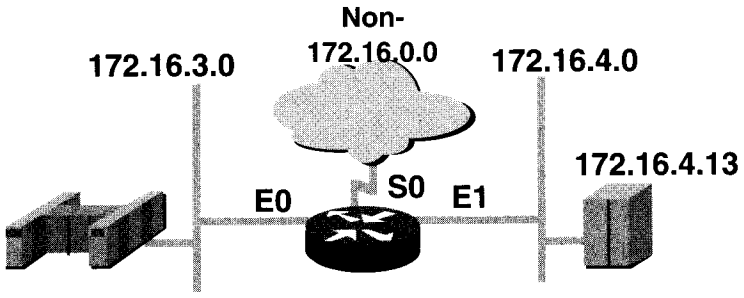
The **access-list** command creates an entry to express a condition statement in a complex filter.

access-list Command	Description
access-list-number	Identifies the list using a number in the range 100 to 199.
permit | deny	Indicates whether this entry allows or blocks the specified address.
protocol	IP, TCP, UDP, ICMP, GRE, IGRP.
source and *destination*	Identifies source and destination IP addresses.
source-mask and *destination-mask*	Wildcard mask; 0s indicate positions that must match, 1s indicate "don't care" positions.
operator and *operand*	lt, gt, eq, neq (less than, greater than, equal, not equal), and a port number.
established	Allows TCP traffic to pass if packet uses an established connection (for example, has ACK bits set).

The **ip access-group** command links an existing extended access list to an outbound interface. Only one access list per port per protocol is allowed.

ip access-group	Description
access-list-number	Indicates the number of the access list to be linked to this interface.
in | out	Selects whether the access list is applied to the incoming or outgoing interface. If **in** or **out** is not specified, **out** is the default.

Extended Access List Example 1




```
access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 21
access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 20
access-list 101 permit ip 172.16.4.0 0.0.0.255 0.0.0.0 255.255.255.255
(implicit deny all)
(access-list 101 deny ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255)

interface ethernet 0
ip access-group 101
```

- **Deny FTP for E0**

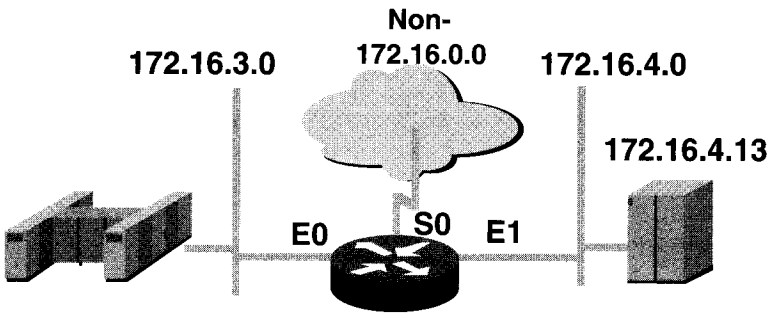
24

In the example:

access-list Command	Description
101	Access list number; indicates extended IP access list.
deny	Traffic that matches selected parameters will be blocked.
tcp	Transport-layer protocol.
172.16.4.0 and *0.0.0.255*	Source IP address and mask; the first three octets must match but do not care about the last octet.
172.16.3.0 and *0.0.0.255*	Destination IP address and mask; the first three octets must match, but do not care about the last octet.
eq 21	Specifies well-known port number for FTP.
eq 20	Specifies the well-known port number for FTP data.
ip access-group 101 Command	Links access list 101 to outgoing port interface E0.

The permit statement allows traffic from subnet 172.16.4.0 to be forwarded to all other networks or subnetworks via interface E0.

Extended Access List Example 2




```
access-list 101 permit tcp 172.16.4.0  0.0.0.255  any eq 25
(implicit deny all)
(access-list 101 deny ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255)

interface ethernet 0
ip access-group 101
```

- **Allow only SMTP for E0**

25

In the example:

access-list Command	**Description**
101	Access list number; indicates extended IP access list.
permit	Traffic that matches selected parameters will be forwarded.
tcp	Transport-layer protocol.
172.16.4.0 and *0.0.0.255*	Source IP address and mask; the first three octets must match but do not care about the last octet.
0.0.0.0 and *255.255.255.255*	Destination IP address and mask; do not care about any octet value.
eq 25	Specifies well-known port number for SMTP.
ip access-group 101 Command	Links access list 101 to outgoing port interface E0.

This example allows only mail from 172.16.4.0 to be sent out interface E0. All other traffic from any other source is denied.

Using Named IP Access Lists

• Feature for Cisco IOS Release 11.2 or newer

Router (config) #

```
ip access-list { standard | extended } name
```

- **Alphanumeric name string must be unique**

Router (config {std- | ext-}nacl) #

```
{ permit | deny } { ip access list test conditions }
{ permit | deny } { ip access list test conditions }
no { permit | deny } { ip access list test conditions }
```

- **Permit or deny statements have no prepended number**
- **"no" removes the specific test from the named access list**

Router (config-if) #

```
ip access-group { name | 1-199 { in | out } }
```

- **Activates the IP named access list on an interface**

26

This feature allows IP simple and extended access lists to be identified with an alphanumeric string (name) instead of the current numeric (1 to 199) representation.

With prior, numbered IP access list statements, an administrator wanting to alter an access list first would be required to delete all the statements in the numbered access list. This deletion uses the word **no** preceding each statement.

Named IP access lists can be used to delete individual entries from a specific access list. This enables you to modify your access lists without deleting and then reconfiguring them. Use named IP access lists when:

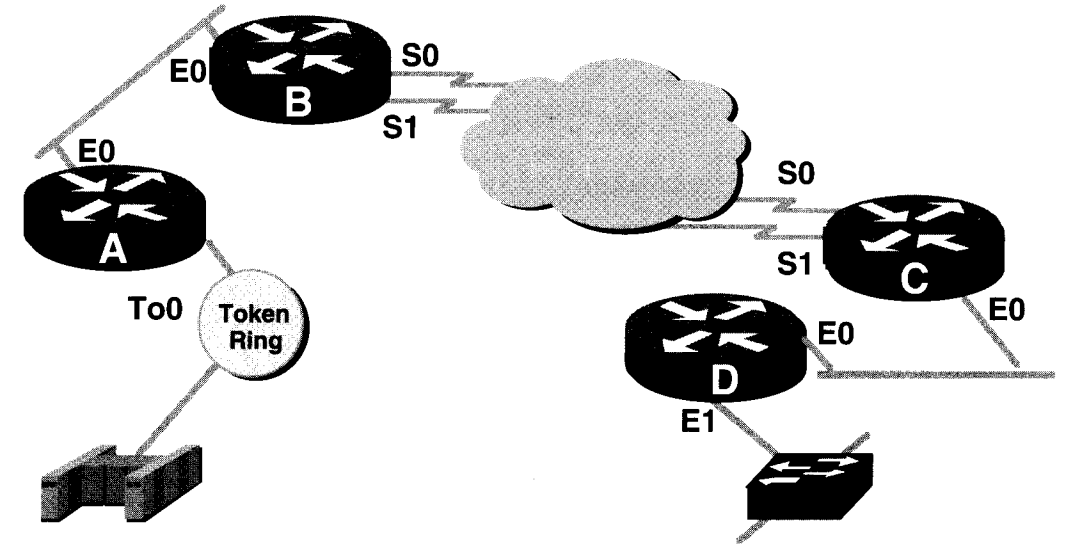
- You want to intuitively identify access lists using an alphanumeric name.
- You have more than 99 simple and 100 extended access control lists to be configured in a router for a given protocol.

Consider the following before implementing named IP access lists:

- Named IP access lists are not compatible with Cisco IOS releases prior to Release 11.2.
- You cannot use the same name for multiple access lists. In addition, access lists of different types cannot have the same name. For example, it is illegal to specify a standard access control list named "George" and an extended access control list with the same name.

Note Most of the commonly used IP access list commands accept named IP access lists.

Where to Place IP Access Lists



- **Place standard access lists close to the destination**
- **Place extended access lists close to the source**

27

An access lists can act as a firewall. A firewall filters packets and eliminates unwanted traffic at a destination. Where the administrator places an access list statement can reduce unnecessary traffic. Traffic that will be denied at a remote destination should not use network resources along the route to that destination.

Suppose an enterprise's policy aims at denying Token Ring traffic on router A to the switched Ethernet LAN on router D's E1 port. At the same time, other traffic must be permitted. Several approaches can accomplish this policy.

The recommended approach uses an extended access list. It specifies both source and destination addresses. Place this extended access list in router A. Then, packets do not cross router A's Ethernet, do not cross the serial interfaces of routers B and C, and do not enter router D. Traffic with different source and destination addresses can still be permitted.

The rule possible with extended access lists is to put the extended access list as close as possible to the source of the traffic denied.

Standard access lists do not specify destination addresses. The administrator would have to put the standard access list as near the destination as possible. For example, place an access list on E0 of router D to prevent traffic from router A.

Monitoring Access Lists

```
Router# show ip interface

Ethernet 0 is up, line protocol is up
     Internet address is 192.54.222.2, subnet mask is 255.255.255.0
     Broadcast address is 255.255.255.255
     Address determined by non-volatile memory
     MTU is 1500 bytes
     Helper address is 192.52.71.4
     Secondary address 131.192.115.2, subnet mask 255.255.255.0
     Outgoing access list 10 is set
     Inbound access list is not set
     Proxy ARP is enabled
     Security level is default
     Split horizon is enabled
     ICMP redirects are always sent
     ICMP unreachables are always sent
     ICMP mask replies are never sent
     IP fast switching is enabled
     Gateway Discovery is disabled
     IP accounting is disabled
     TCP/IP header compression is disabled
     Probe proxy name replies are disabled
Router#
```

28

The **show ip interface** command displays IP interface information and indicates whether any access lists are set.

Monitoring Access List Statements

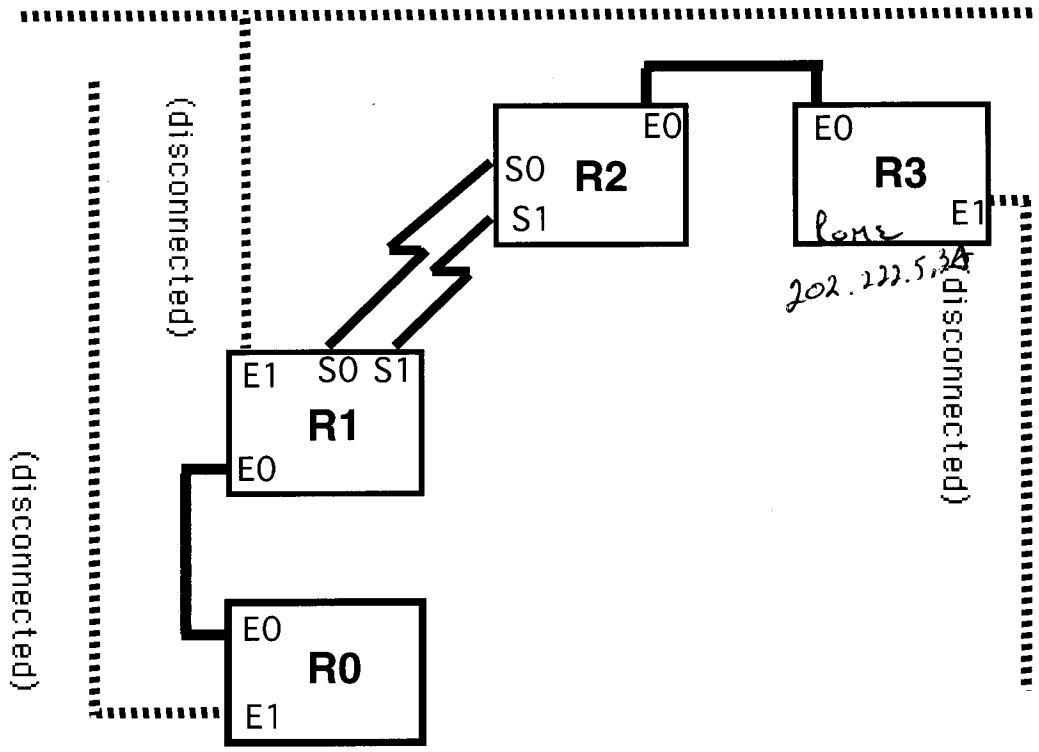
```
Router> show access-lists

Standard IP access list 19
     permit 172.16.19.0
     deny   0.0.0.0, wildcard bits 255.255.255.255
Standard IP access list 49
     permit 172.16.31.0, wildcard bits 0.0.0.255
     permit 172.16.194.0, wildcard bits 0.0.0.255
     permit 172.16.195.0, wildcard bits 0.0.0.255
     permit 172.16.196.0, wildcard bits 0.0.0.255
     permit 172.16.197.0, wildcard bits 0.0.0.255
Extended IP access list 101
     permit tcp 0.0.0.0  255.255.255.255  0.0.0.0  255.255.255.255 eq 23
Type code access list 201
     permit 0x6001 0x0000
Type code access list 202
     permit 0x6004 0x0000
     deny   0x0000 0xFFFF
Router>
```

29

The **show access-lists** command displays the contents of all access lists. This Cisco IOS command provides more details about the access list statements. By entering the access list name or number as an option for this command, you can see a specific list.

Lab: Standard and Extended Access Lists

Standard Access Lists Data Sheet



Objective: Configure standard and extended access lists to filter IP traffic.

Objective: Monitor and verify selected access list operations on the router.

Instructions: Verify that basic packet filtering only looks at the IP address layer. Use standard access lists to prevent packets from flowing between R0 and R3 and between R3 and R0.

Step 1 Use a **show access-list** command to make sure that no earlier IP access lists are still active on your router interfaces.

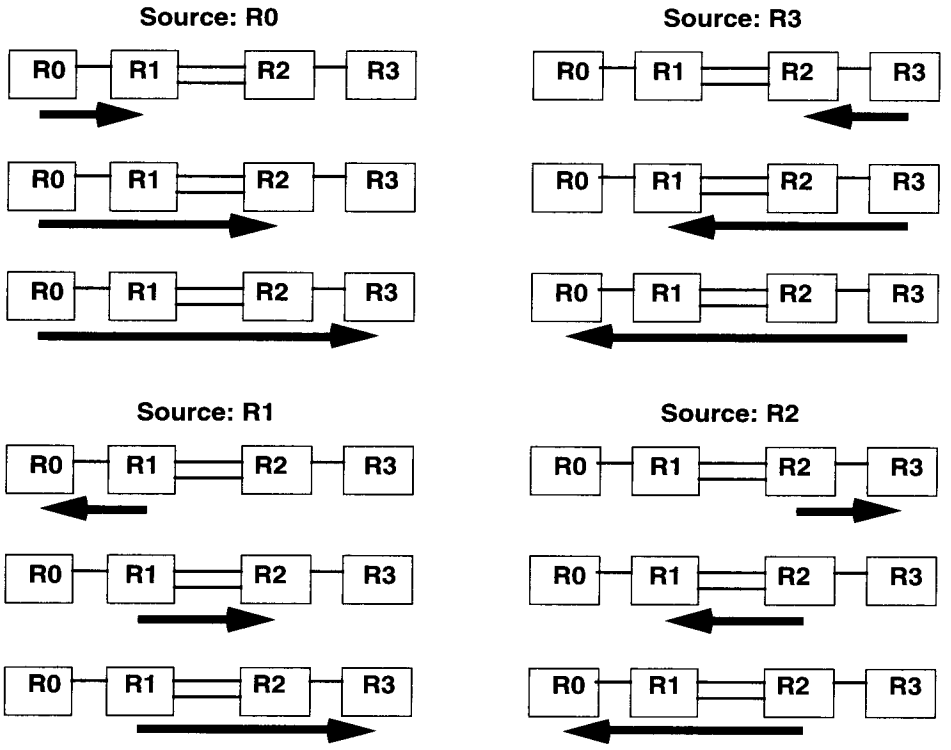
Step 2 Use **ping** to test connectivity between R0 and R3, and between R3 and R0.

Step 3 You will work in teams within your group.

Step 4 Check the E1 connections on R0, R1, and R3 to make sure there is no alternate path between R0 and R3. The E1 interfaces should be shut down.

Standard Access Lists Implementation

00160000 00160111



Instructions: Identify where to enter standard access lists to prevent packets from flowing between R0 and R3 and between R3 and R0. Enter the access lists and verify their function.

Step 1 Answer the following questions by drawing circles in the appropriate locations in the diagram. Reach agreement with your team members on these important issues before proceeding.

- Which traffic will be filtered?
- Which routers will be configured with an access list?
- Which interface(s) will receive the access list?

Step 2 Design the appropriate access list to prevent packets flowing between R0 and R3. Give your access list a valid number.

Step 3 Apply the access group to specific outgoing interface(s).

Step 4 Use the **show access-list** command to check the access list and **show ip interface** to check the interface to which the access list is attached.

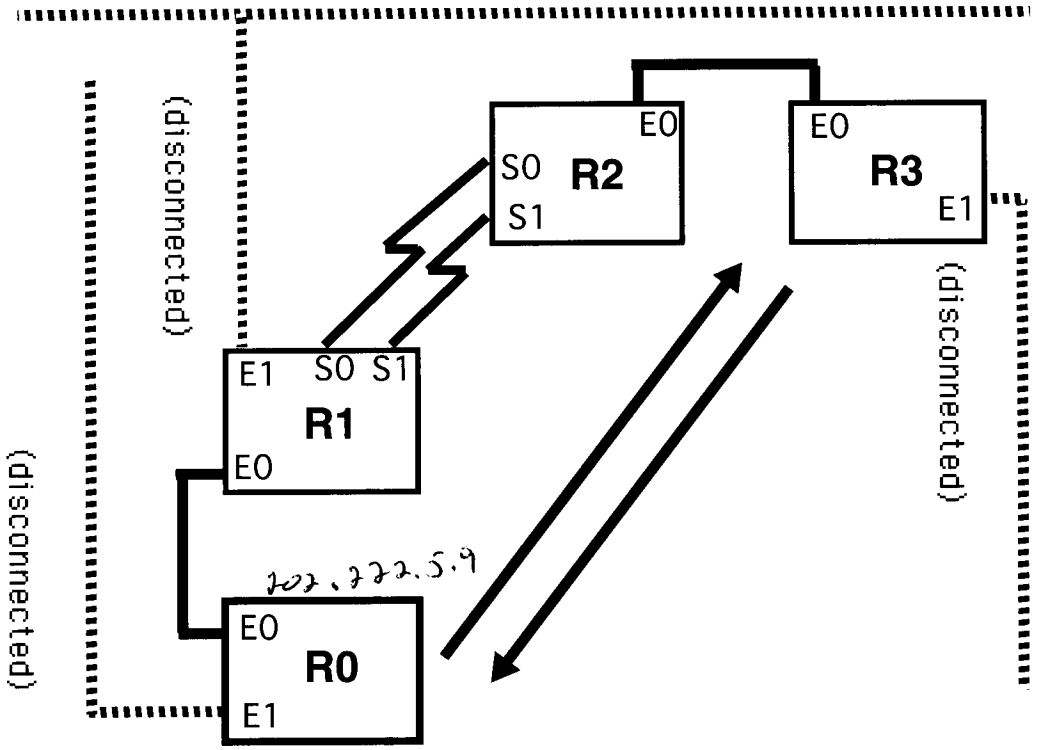
Step 5 Use **ping** to check results. R0 and R3 should be able to reach R1 and R2 successfully, but access should be denied between R0 and R3.

Step 6 As soon as you have achieved this lab objective, use the **no** option to remove each of your standard IP access list commands from your running configuration. Save the results in your startup-config.

202.222.5.32 202.222.5.39

00.

IP Extended Access List



Instructions: Design, deploy, and observe the effect of extended IP access lists on routers to control connectivity and to reduce network congestion. Design the required access lists to prevent Telnet traffic from flowing from R0 to R3, and from R3 to R0. Allow SMTP and ICMP messages between these same routers. Apply the access lists on the appropriate interfaces within your workgroup. Test for the desired connectivity and observe that only the "prevented" traffic is being denied.

Step 1 Using the **ping** and **telnet** commands, ensure that full connectivity exists within your workgroup.

Step 2 As a group, design the appropriate access list(s) and assign them to the appropriate interfaces.

Step 3 Use the command **show access-list** to check that the access lists have been installed.

Step 4 Use **telnet** and **ping** to check the results. Do they both work? Only one? Neither? What would you expect to happen? Are you getting the desired results?

Step 5 Remove the access lists from the routers that have them and verify workgroup connectivity.

IP Access-list 101 DENY TCP 202.222.5.9 0.0.0.0 eq 23

IP Access-list 101 PERMIT IP Any Any

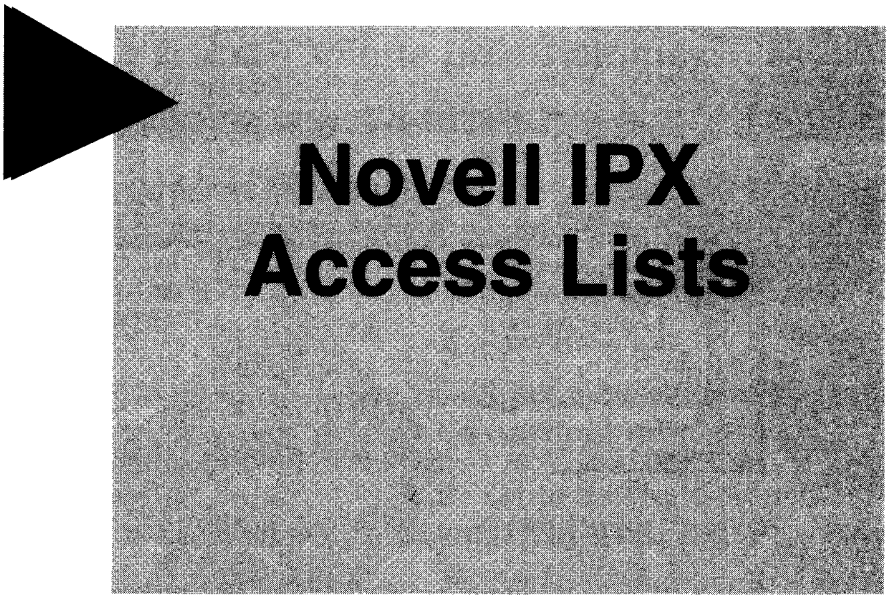
Then. Apply List to Interface

Reserved TCP Port Numbers

Decimal	Keyword	Description
0		Reserved
1-4		Unassigned
5	RJE	Remote Job Entry
7	ECHO	Echo
9	DISCARD	Discard
11	USERS	Active Users
13	DAYTIME	Daytime
15	NETSTAT	Who is Up or NETSTAT
17	QUOTE	Quote of the Day
19	CHARGEN	Character Generator
20	FTP-DATA	File Transfer Protocol (data)
21	FTP	File Transfer Protocol
23	TELNET	Terminal Connection
25	SMTP	Simple Mail Transfer Protocol
37	TIME	Time of Day
39	RLP	Resource Location Protocol
42	NAMESERVER	Host Name Server
43	NICNAME	Who Is
53	DOMAIN	Domain Name Server
67	BOOTPS	Bootstrap Protocol Server
68	BOOTPC	Bootstrap Protocol Client
69	TFTP	Trivial File Transfer Protocol
75		Any Private Dial-out Service
77		Any Private RJE Service
79	FINGER	Finger
95	SUPDUP	SUPDUP Protocol
101	HOSTNAME	NIC Host Name Server
102	ISO-TSAP	ISO-TSAP
113	AUTH	Authentication Service
117	UUCP-PATH	UUCP Path Service
123	NTP	Network Time Protocol
133-159		Unassigned
160-223		Reserved
224-241		Unassigned
242-255		Unassigned

Reserved UDP Port Numbers

Decimal	Keyword	Description
0		Reserved
1–4		Unassigned
5	RJE	Remote Job Entry
7	ECHO	Echo
9	DISCARD	Discard
11	USERS	Active Users
13	DAYTIME	Daytime
15	NETSTAT	Who is Up or NETSTAT
17	QUOTE	Quote of the Day
19	CHARGEN	Character Generator
20	FTP-DATA	File Transfer Protocol (data)
21	FTP	File Transfer Protocol
23	TELNET	Terminal Connection
25	SMTP	Simple Mail Transfer Protocol
37	TIME	Time of Day
39	RLP	Resource Location Protocol
42	NAMESERVER	Host Name Server
43	NICNAME	Who Is
53	DOMAIN	Domain Name Server
67	BOOTPS	Bootstrap Protocol Server
68	BOOTPC	Bootstrap Protocol Client
69	TFTP	Trivial File Transfer Protocol
75		Any Private Dial-out Service
77		Any Private RJE Service
79	FINGER	Finger
123	NTP	Network Time Protocol
133-159		Unassigned
160-223		Reserved
224-241		Unassigned
242-255		Unassigned

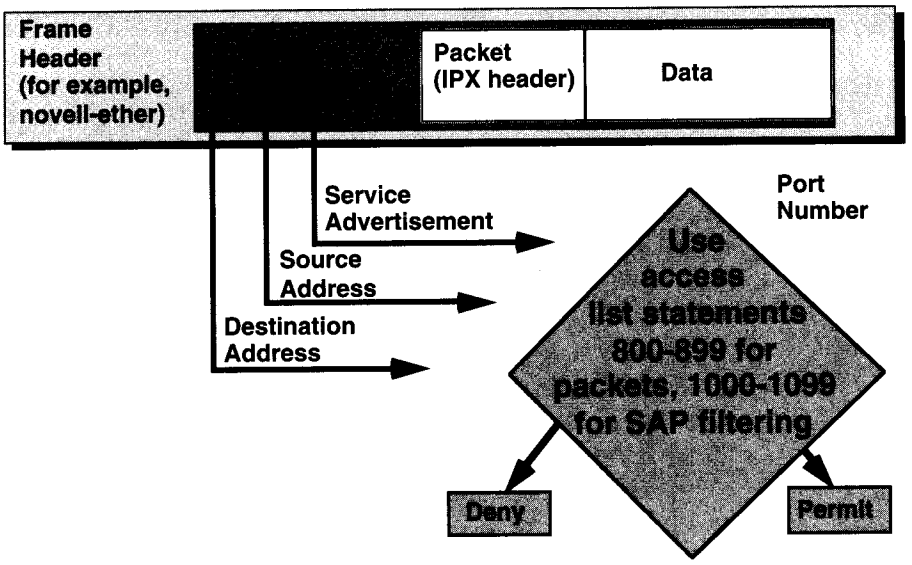


35

Novell IPX Access Lists

Testing Packets with Access Lists

An Example Using an IPX Packet



36

For the Novell IPX packet filters covered in this chapter, Cisco IOS access lists check the packet header for:

- Destination and source IPX addresses using standard access lists; identify these with a number in the range 800 to 899.
- Service advertisement numbers in addition to the other tests in SAP filter access lists; identify these with a number in the range 1000 to 1099.

For all of these Novell IPX access lists, after a packet is checked for a match with the access list statement, it can be denied or permitted to use an interface in the access group.

Note Cisco IOS offers several other forms of access lists for Novell IPX packets. Refer to the Cisco Connection Documentation, Enterprise Series CD-ROMs for further information.

Key Concepts for IPX Access Lists

- **IPX addressing uses a network.node and a socket number**
- **Standard lists (800 to 899) can filter source and destination address**
- **Access lists (1000 to 1099) are SAP filters for service types and servers on one or more networks**
- **Other access list number ranges offer additional Novell software filters (examples: GNS, RIP, NLSP)**

37

Novell addressing is based on network.node.socket. The network number is assigned by the administrator; the node portion is derived from the MAC address of the individual interface. Serial lines adopt the MAC address of another interface in the creation of their logical addresses. The socket number refers to a process or application (somewhat like the TCP segment).

Every NetWare file server has an internal IPX network number and performs IPX routing. External IPX networks attach to router interfaces. The IPX network number assigned on a Cisco router's interface must be unique and consistent with the network numbers known to the file server.

IPX standard access lists use numbers in the range 800 to 899. These access lists check for either source address or both source and destination address. To identify parts of the address to check or ignore, IPX standard access lists use a wildcard mask that operates like the mask used with IP addresses.

To control the traffic from the Service Advertisement Protocol (SAP), use SAP filters that use numbers in the range 1000 to 1099. Several other packet and route filters can help manage IPX overhead traffic. For example, access lists can control Get Nearest Server (GNS) from clients to servers, Routing Information Protocol (RIP), and NetWare Link Services Protocol (NLSP).

IPX Standard Access List Configuration

Router (config) #

access-list *access-list-number* { deny | permit } *source-network*
[*.source-node*] [*source-node-mask*] [*destination-network*]
[*.destination-node*] [*destination-node-mask*]

- **Sets parameters for this list entry**
- **Standard acccess list uses list-number in range 800 to 899**

Router (config-if) #

ipx access-group *access-list-number*

- **Activates the IPX standard access list on an interface**

38

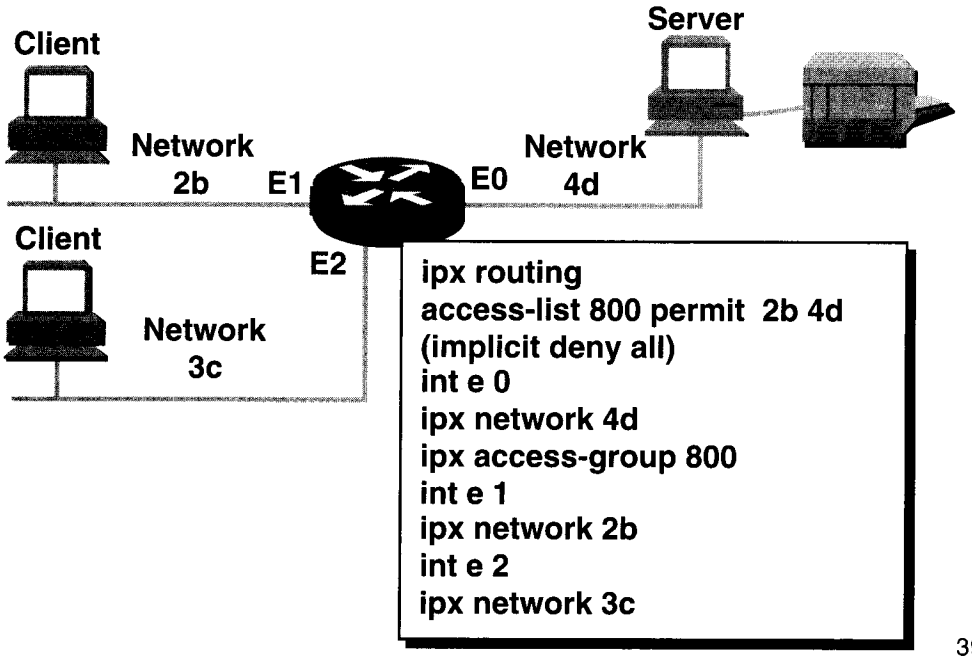
Use the **access-list** command to filter traffic in an IPX network. Using filters on the outgoing router interface allows or restricts different protocols and applications on individual networks.

access-list Command	Description
access-list-number	Access list number for an IPX filter list from 800 to 899.
protocol EXTENDED ACL ONLY	Number of the protocol type, can be: 0=any protocol (refer to socket number below), 1=RIP, 4=SAP, 5=SPX, 17=NCP, 20=IPX NetBIOS.
source-network	Source network number, expressed in eight-digit hexadecimal.
source-node	Node number on the source network. Represented as a 48-bit value shown in a dotted triplet of 4-digit hexadecimal numbers.
destination-network	Network number to which the packet is being sent.
destination-node	Node on the destination network to which the packet is being sent.

Use the **ipx access-group** command to link an IPX traffic filter to an interface.

ipx access-group Command	Description
access-list-number	Access list number for an IPX filter list from 800 to 899.

Standard IPX Access List Example



39

In the example:

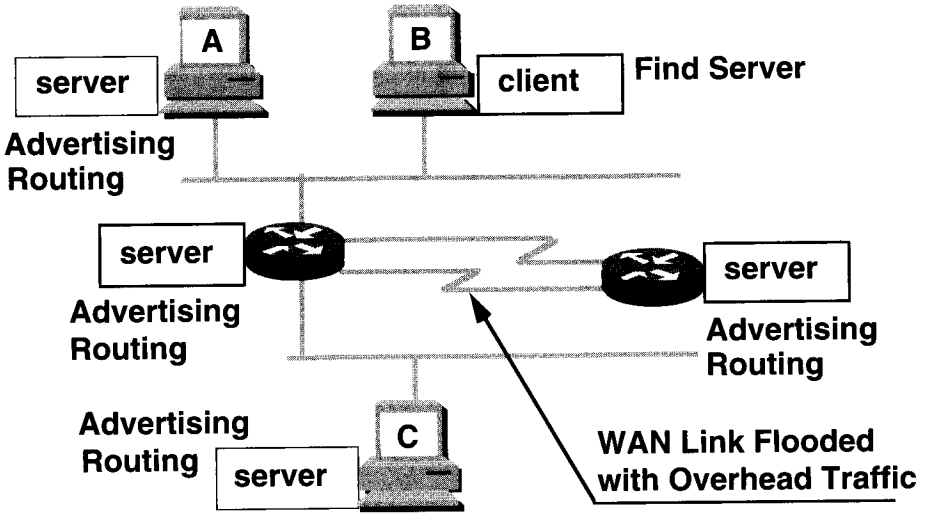
access-list 800 permit 2b 4d Command	**Description**
800	Specifies a Novell IPX standard access list.
permit	Traffic matching the selected parameters will be forwarded.
2b	Source network number.
4d	Destination network number.
(implicit deny all)	Not a valid configuration command, just a reminder that access lists filter traffic not specified to be forwarded.
ipx access-group 800 Command	Links access list 800 to outgoing interface E0.

Traffic from network 2b destined for network 4d will be forwarded out Ethernet 0.

The access list is applied to an outgoing interface and filters outbound packets.

Notice that the other interfaces E1 and E2 are not subject to the access list; they lack the access group statement to link them to the access list 800.

Why to Control IPX Overhead



- **Frequent updates reduce the bandwidth for user traffic**

40

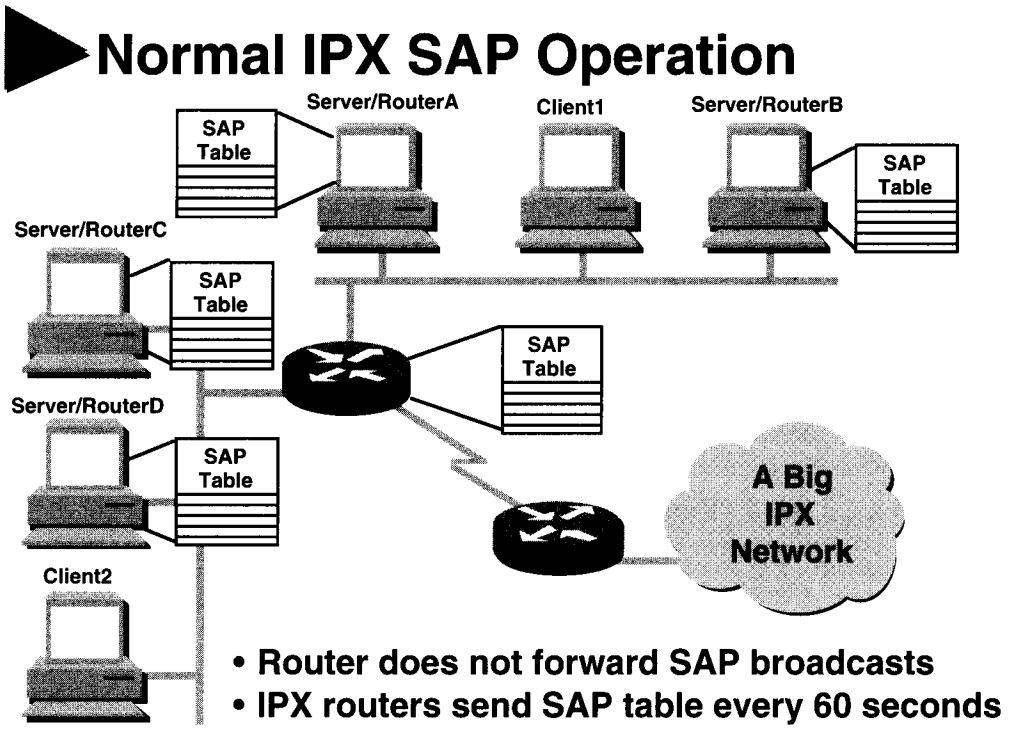
IPX routing and advertising processes were developed to run on LANs. As LANs interconnect with slower, more costly WAN links, concerns increase about the traffic with overhead from IPX control packets reducing the bandwidth available for user applications traffic.

IPX servers broadcast service advertising (SAPs) details every 60 seconds.

Routers broadcast routing information and metrics to other IPX routers. The graphic shows four IPX routers: the two servers A and C, as well as the two Cisco routers.

Whenever a client workstations starts up, it sends its own SAP broadcast to find a server; then from the nearest server, the client can log in to a target server and run network applications from network drives.

Whenever packets from these protocols are unwanted, a network administrator can set up IPX access lists. With the standard access lists in this chapter, the permit/deny filtering acts on all IPX packets for the interface addresses.



SAP broadcasts synchronize the list of available services. The NetWare files server acts like an IPX router. The Cisco router acts like a SAP server.

If the router passed a SAP every time it received one, the WAN link would be flooded with SAP traffic. The router will not forward SAP broadcasts.

Instead, both file servers and routers listen to SAP messages and build a SAP table. All devices that build SAP tables advertise this information every 60 seconds.

This can still result in considerable overhead as all these servers and routers send their own complete SAP table every 60 seconds.

How to Use SAP Filters

SAP Filter Goals

deny type 7 (print server) SAP from 2a
deny type 98 (access server) SAP from 5b
deny type 24 (router) SAP to 7c
deny type 4 (file server) SAP from 4a
deny type 26a (NMS)
deny type 7a (NetWare for VMS) from *8
permit the remaining SAPs

- **Plan for SAP filters and enter global command**
- **Note: Must set up SAP filters on all routers**

42

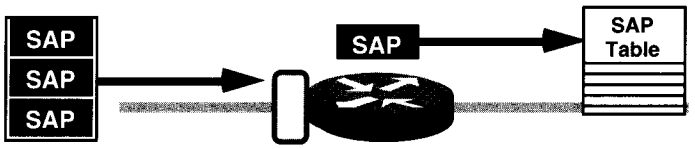
You must carefully plan for SAP filtering before configuring. Make sure that all clients will see advertisements necessary and sufficient for their application processing. You will need to enter the SAP filters in any and all routers where you want them to operate. A table of the most common SAP numbers follows.

SAP Number	Server Type
4	NetWare file server
7	Print server
24	Remote bridge server (router)

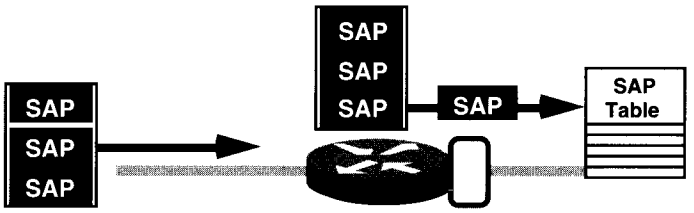
Place SAP filters close to the source. Proper placement of SAP filters conserves critical bandwidth, especially on serial links.

How to Use SAP Filters (cont.)

Input filter: Do not add filtered SAPs to SAP table



Output filter: Do not add filtered SAPs to the SAP table sent



- **Apply the access list to the interface as an input or output SAP filter**

43

When a SAP advertisement arrives at the router interface, the contents are placed in the SAP table portion of main memory. The contents of the table are propagated during the next SAP update.

Two types of access list filters control SAP traffic:

- IPX input SAP filter

  When a SAP input filter is in place, the services entered into the SAP table are reduced. The propagated SAP updates represent the entire table, but contain only a subset of all services.

- IPX output SAP filter

  When a SAP output filter is in place, the services propagated from the table are reduced. The propagated SAP updates represent a portion of the table contents and are a subset of all the known services.

SAP Filter Configuration

Router (config) #

access-list ***access-list-number*** { **deny** | **permit** } ***network*** [***.node***] [***network-mask node-mask***] [***service-type*** [***server-name***]]

- **Creates an entry in a SAP filter list**

Router (config-if) #

ipx input-sap-filter ***access-list-number***

- **Activates the input SAP filter on the interface**

or

Router (config-if) #

ipx output-sap-filter ***access-list-number***

- **Activates the output SAP filter on the interface**

44

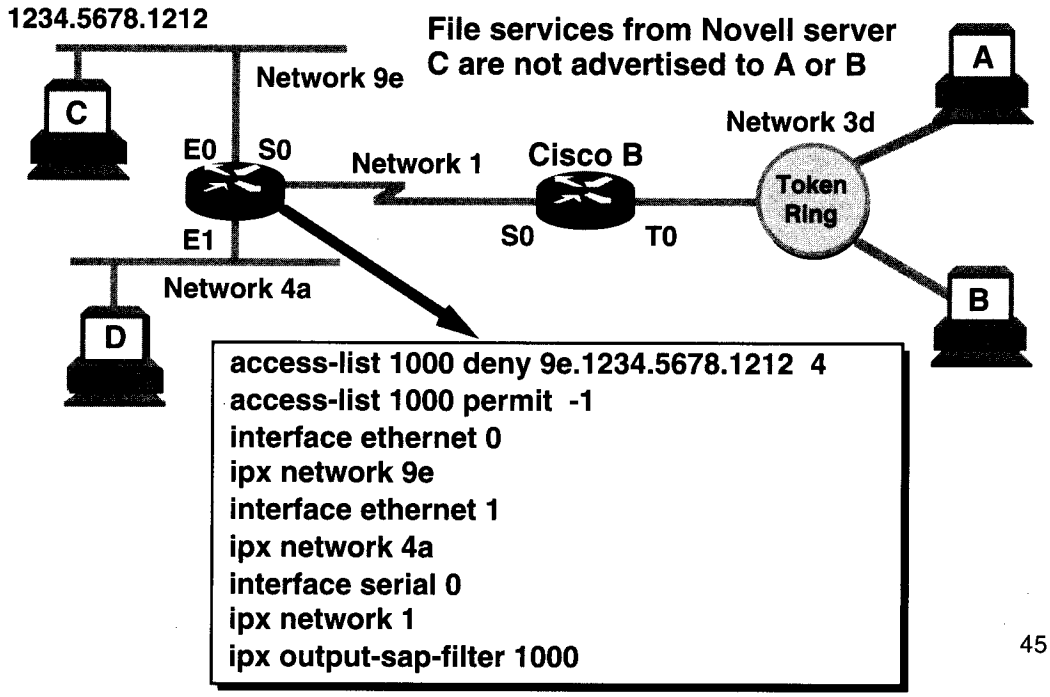
Use the **access-list** command to control propagation of the SAP messages.

access-list Command	**Description**
access-list-number	Number from 1000 to 1099, indicates a SAP filter list.
network [*.node*]	Novell source internal network number with optional node number; -1 is all networks.
network-mask node-mask	Mask to be applied to the network and node. Place ones in the positions to be masked.
service-type	SAP service type to filter. Each SAP service type is identified by a hexadecimal number. Some common examples are:
4	File server.
7	Print server.
24	Remote bridge server (router).
server-name	Name of the server providing the specified service type.

The **ipx input-sap-filter** and **ipx output-sap-filter** commands place a SAP filter on an interface. The use of **input** or **output** determines whether SAPs are filtered before entry into the SAP table, or whether the SAP table contents are filtered during the next update.

SAP table content can be filtered on input by using the **ipx router-sap-filter** command, which identifies from which router SAP advertisements can be received.

SAP Filter Example 1



45

In the example:

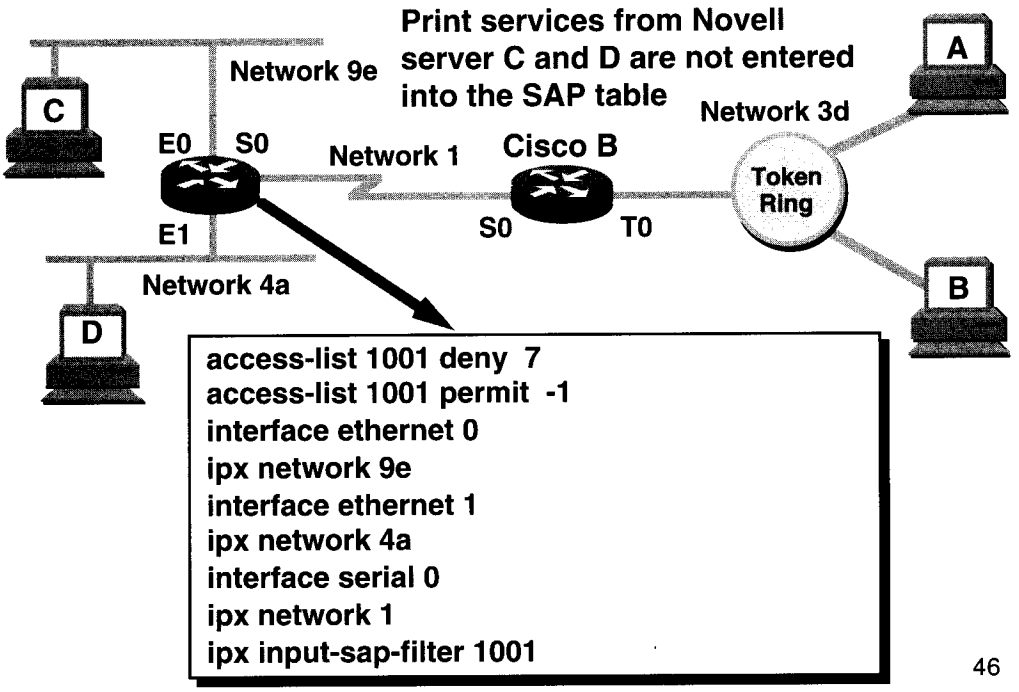
access-list 1000 deny 9e. 1234.5678.1212 4 Command	**Description**
1000	An access list number in the Novell SAP filter range.
deny	SAP services matching selected parameters will be blocked.
9e.1234.5678.1212	Source network address of SAP advertisement.
4	Type of SAP service; advertises file service.

access-list 1000 permit -1 Command	**Description**
1000	Access list number.
permit	SAP services matching parameters will be forwarded.
-1	Source network number; -1 means all networks.

ipx output-sap-filter 1000 Command	Places list 1000 on interface serial 0 as an output SAP filter.

File server advertisements from server 9e.1234.5678.1212 will not be forwarded on interface serial 0 (S0). All other SAP services from any source will be forwarded on interface S0.

SAP Filter Example 2



46

In the example:

access-list 1001 deny 7 **Command**	**Description**
1001	An access list number in the Novell SAP filter range.
deny	SAP services matching selected parameters will be blocked.
7	Type of SAP service; advertises print service.

access-list 1001 permit -1 **Command**	**Description**
1001	Access list number.
permit	SAP services matching parameters will be forwarded.
-1	Source network number; -1 means all networks.

ipx input-sap-filter 1001 **Command**	Places list 1001 on interface serial 0 as an input SAP filter.

Print server advertisements from servers C and D will not be entered into the SAP table. All other SAP services from any source will be added into the SAP table.

Monitoring IPX Access Lists

```
dtp-19#sh ipx int et1/1
Ethernet1/1 is up, line protocol is up
  IPX address is 10.0000.0c0d.724f, NOVELL-ETHER [up]
  Delay of this IPX network, in ticks is 1 throughput 0 link delay 0
  IPXWAN processing not enabled on this interface.
  IPX SAP update interval is 1 minute(s)
  IPX type 20 propagation packet forwarding is disabled
  Incoming access list is not set
  Outgoing access list is not set
  IPX helper access list is not set
  SAP GNS processing enabled, delay 0 ms, output filter list is not se
  SAP Input filter list is not set
  SAP Output filter list is not set
  SAP Router filter list is not set
  Input filter list is 800
  Output filter list is 801
  Router filter list is not set
  Netbios Input host access list is not set
  Netbios Input bytes access list is not set
  Netbios Output host access list is not set
  Netbios Output bytes access list is not set
  Updates each 60 seconds, aging multiples RIP: 3 SAP: 3
  SAP interpacket delay is 55 ms, maximum size is 480 bytes
  RIP interpacket delay is 55 ms, maximum size is 432 bytes
  IPX accounting is disabled
  IPX fast switching is configured (enabled)
  RIP packets received 28460, RIP packets sent 24999
  SAP packets received 4, SAP packets sent 2
```

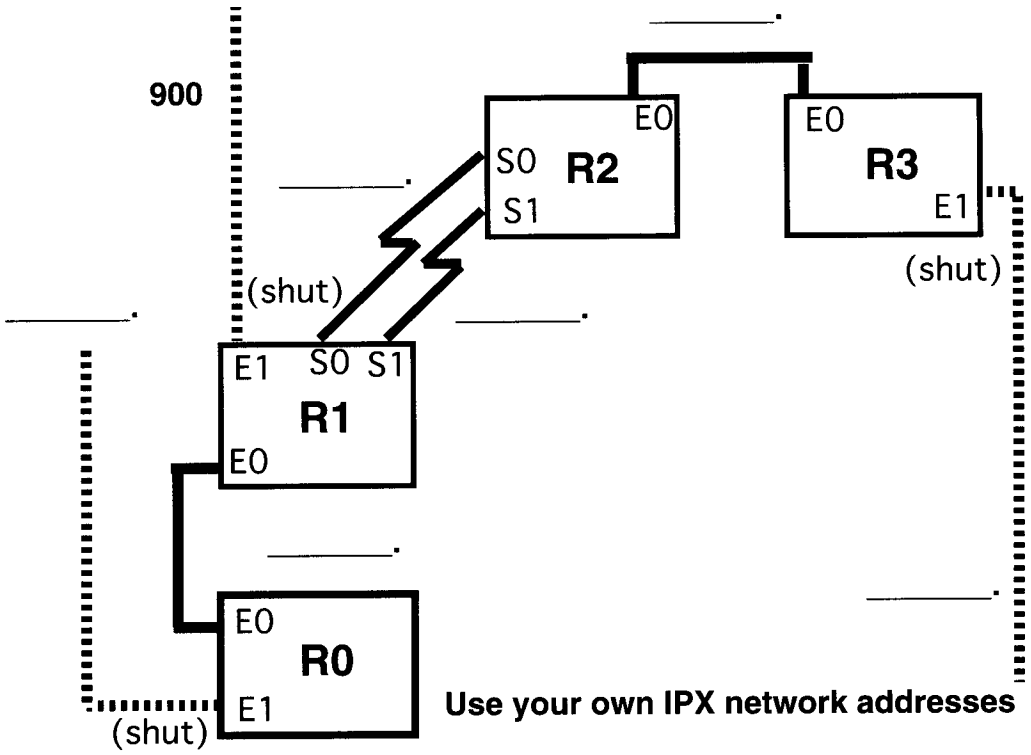
```
dtp-19#sh access-lists
IPX access list 800
    deny 8000
IPX access list 801
    deny FFFFFFFF
```

47

The **show ipx interface** command displays information about the configuration of the interface. It shows that the input filter list is 800 and the output filter list is 801. The **show access-lists** command displays the contents of lists 800 and 801.

Lab: Novell IPX Access Lists

Novell Standard IPX Access Lists Data Sheet



Objective: Configure IPX access lists and SAP filters to control basic Novell traffic.

Instructions: Write the Cisco IOS commands to configure IPX access lists filters to control basic Novell traffic on the lines provided. Use this page for planning.

Step 1 Refer to the IPX network addresses you used in when you configured IPX for your group. Write the router interface addresses onto the graphic. The IPX network addresses for the E1 interface on router R1 is network number 900.

Step 2 Use extended **ping** and make sure of IPX connectivity between R0, R1, R2, and R3.

Step 3 Shut the E1 interfaces on R0, R1, and R3 to make sure there is no alternate path between R0 and R3.

Novell IPX Access List Implementation

Instructions: Refer to the diagram and information on the Novell IPX Access Lists Data Sheet. Use standard access lists to prevent packets from flowing between R0 and R3 and between R3 and R0.

Do this lab with the other members of your router group. Decide which router or routers should perform each step and only perform a step on your router if appropriate.

Step 1 Below enter standard IPX access list statements to prevent any NetWare client on the network on router R3's E0 interface from accessing the network on router R0's E0 interface.

Check the statements with your group to reach consensus. Then, if appropriate, enter the agreed-upon access list statements into your router configuration.

Step 2 Enter standard IPX access list statements to prevent any NetWare client on the network on router R0's E0 interface from accessing the network on router R3's E0 interface.

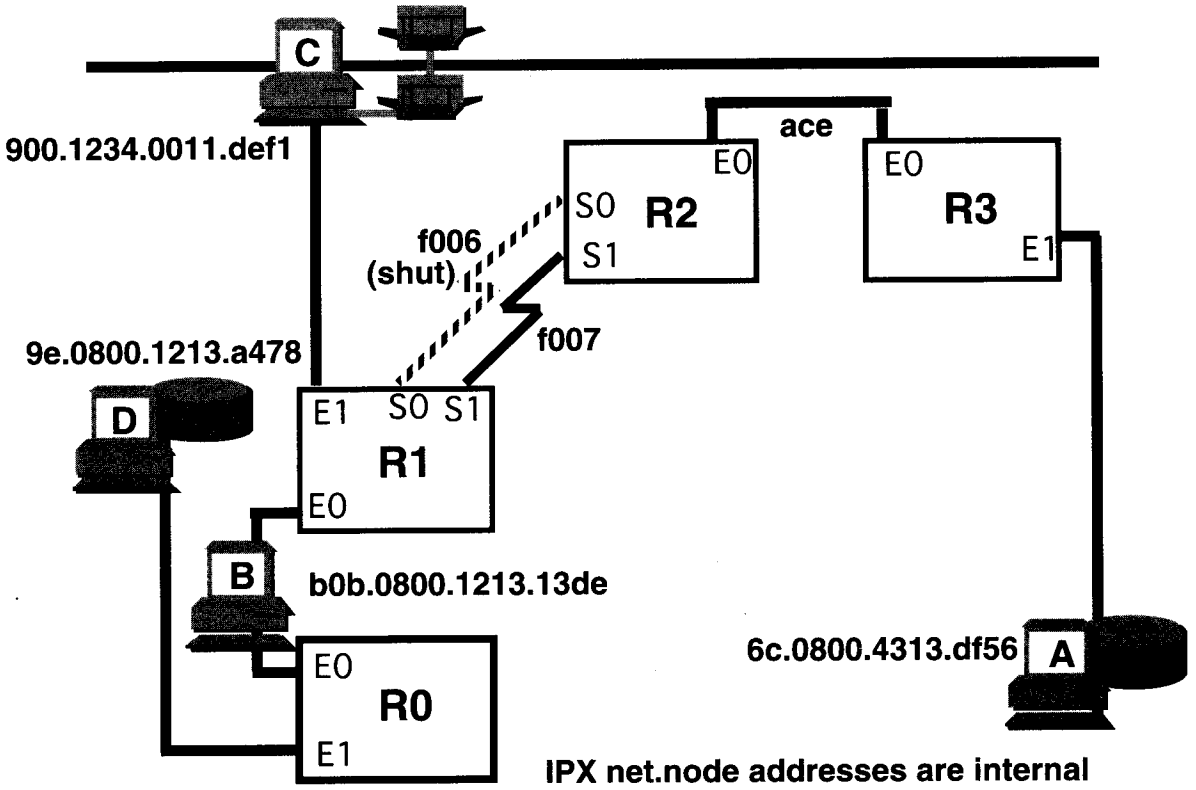
Check the statements with your group to reach consensus. Then, if appropriate, enter the agreed-upon access list statements into your router configuration.

Step 3 Use the **show ipx interface** command to monitor your IPX standard access lists. Compare the output to the graphic showing your group.

Step 4 Use the extended **ping** command to test access within your group. Verify that the standard IPX access lists accomplish the control policy described in steps 1 and 2.

Step 5 When your group agrees that the access lists and SAP filters appear to be correct, remove the shuts on the E1 interfaces on R0, R1, and R3.

Exercise: Novell IPX SAP Filters Data Sheet



Instructions: Write (but do not enter into your router) the Cisco IOS commands that would configure IPX SAP filters to control specified Novell overhead traffic. For this exercise, assume that your IPX internetwork includes the following IPX server types:

File Server (type 4)

Print Server (type 7)

Access Server (type 98)

Use the IPX network example and solve the four problems shown on the next pages.

Exercise: Novell IPX SAP Filters Configuration

Objective: Configure IPX access lists and SAP filters to control basic Novell traffic.

Instructions: Write the commands used to configure specified IPX SAP filters.

Refer to the diagram and information on the Novell IPX SAP Filters Data Sheet. Use SAP filters to limit server advertisement processing. In the diagram, A and D are NetWare file servers. B is a Novell access server, and C is a print server. Routers R0, R1, R2, and R3 are considered remote-bridge servers for IPX.

Work as a group to determine the configuration to filter SAP packets as follows.

For This Traffic	Apply This Filtering
Print services of C	Are not sent to A
File services of D	Are not sent to R2 or R3
Services of A	Are not sent to C
All SAPs from B	Are not sent over serial links

Problem 1

Enter the commands to configure a SAP filter to prevent the print services from server C from going to A. Check the statements with your group to reach consensus. Indicate on which router(s) you would configure the SAP filters, then write the agreed-upon access list statements.

Problem 2

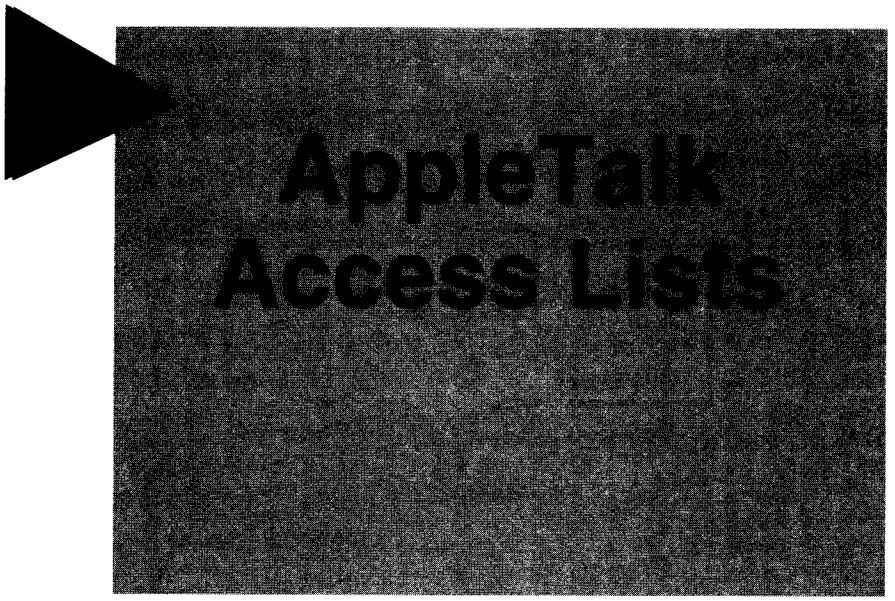
Enter the the commands to configure a SAP filter to prevent the file services from server D from going to routers R2 and R3. Check the statements with your group to reach consensus. Indicate on which router(s) you would configure the SAP filters, then write the agreed-upon access list statements.

Problem 3

Enter the SAP filter to prevent the services from server A from going to server C. Check the statements with your group to reach consensus. Indicate on which router(s) you would configure the SAP filters, then write the agreed-upon access list statements.

Problem 4

Enter the SAP filter to prevent all SAPs from server B from going out R1's serial lines. Check the statements with your group to reach consensus. Indicate on which router(s) you would configure the SAP filters, then write the agreed-upon access list statements.

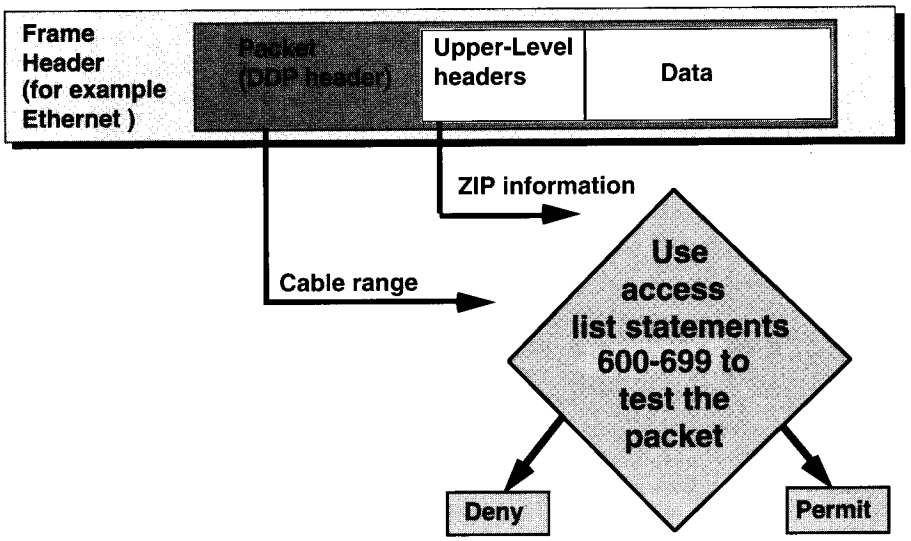


53

AppleTalk Access Lists

Testing Packets with Access Lists

An Example Using an AppleTalk Packet



54

For the AppleTalk packet filters covered in this chapter, Cisco IOS access lists check the packet header for:

- Cable range or network numbers with access lists; identify these with a number in the range 600 to 699.
- Zone Information Protocol (ZIP) replies with zip-reply-filter access lists; also identify these with a number in the range 600 to 699.

For all of these AppleTalk access lists, after a packet is checked for a match with the access list statement, it can be denied or permitted to use an interface in the access group.

Note Cisco IOS offers several other forms of access lists for AppleTalk packets. Refer to the Cisco Connection Documentation, Enterprise Series CD-ROMs for further information.

Key Concepts for AppleTalk Access Lists

- **AppleTalk lists (600 to 699) offer several packet filters**
- **Filter extended networks or cable range**
- **Select partial cable range filters within extended networks**
- **Zones divide networks into communities of interest**
- **Use lists to limit ZIP traffic**

55

A key AppleTalk concept hides network numbering from end users. End users may see zones and resources, but numerical configuration is a hidden issue for the network administrator.

Administrators can use AppleTalk filters to control traffic by referring to the 16-bit network number portion of a full 24-bit address. Because the node portion is dynamically assigned as AppleTalk nodes come up, these node numbers are not predictable for access list entries.

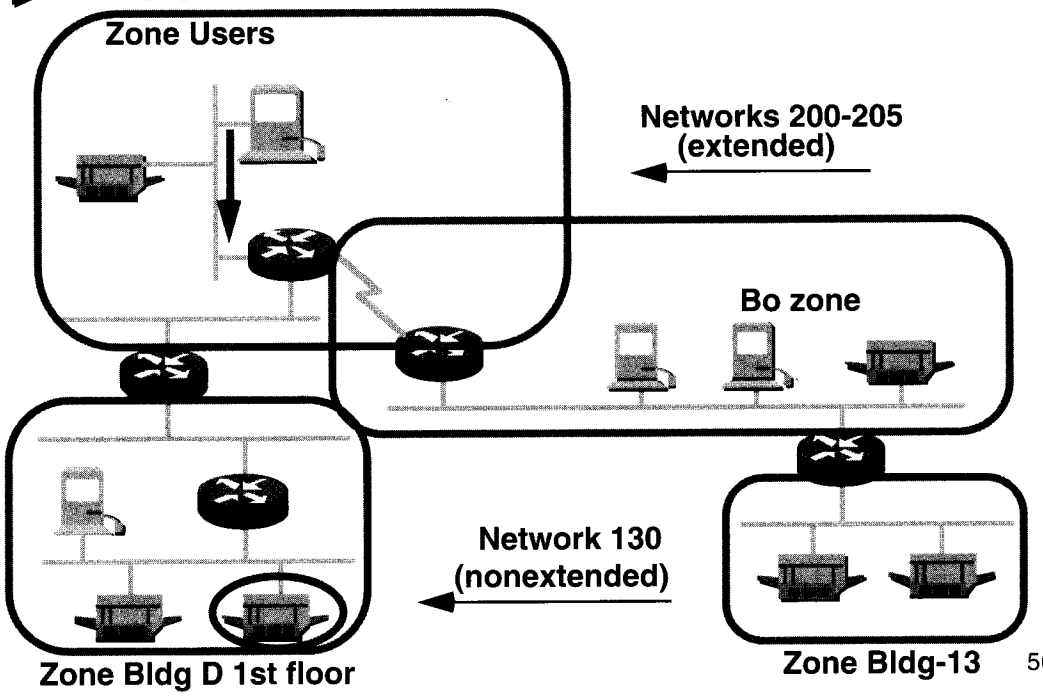
Although earlier AppleTalk networks offered a single nonextended network on a single medium, current AppleTalk uses extended addressing. This means that more than one AppleTalk network can occupy the same physical media. Express one or more AppleTalk networks on the medium as the cable range.

An administrator can filter an entire cable range.

Alternately, by including the term "within," an administrator can select portions of a cable range for access list testing. One use of a partial cable range is when an Appletalk administrator establishes a broad cable-range for an interface to a location (say a remote regional office), then wants to identify subsets of the cable-range for the various departments at the regional office. The administrator specifies access-list filters appropriate to the different departments. Then access to the interface can be permitted or denied within the cable-range subsets appropriate for each of the departments.

ZIP filters are one method for reducing AppleTalk zone information update distribution traffic.

AppleTalk Network Structures



It has become commonplace for routed AppleTalk networks to evolve into complex internetworks. As growth extends across LANs and serial lines, access list controls involve several AppleTalk network structures.

The first is the grouping of networks and their resources into zones. These are arbitrary subsets of nodes within the AppleTalk internetwork. One zone called Users contains a separate group of resources from those in zones Bldg D 1st floor and Bldg-13.

Current AppleTalk internetworks use extended network addresses. For example, an Ethernet transmission medium in zone Users can contain networks in the contiguous network number range of 200 to 205. Older internetworks continue using nonextended network addressing such as 130 in Bldg D 1st floor.

The user's application sends output to the print manager. For network access, the routing tables and zone information helps direct the user's output from its source to the destination zone containing the selected printer. The administrator can use access lists to control traffic based on network and cable-range selections.

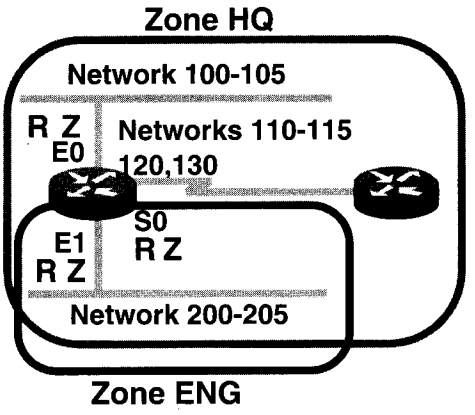
AppleTalk Access List Procedures

Access list configuration

- AppleTalk list numbers: 600-699
- Specify permit or deny access
- Enter source net or cable range

Access group configuration

- Apply list number to interface
- Filter data or specify overhead packets



R = AppleTalk RTMP

Z = AppleTalk ZIP

57

To configure for AppleTalk number access lists, select a unique access list number from within the range 600 to 699.

As with the other protocols, the AppleTalk access list statement requires a permit or deny in each statement to specify traffic controls to potential outgoing interfaces.

With phase 1 addressing, specify a single network number such as network130. More commonly, specify phase 2 addresses by entering a cable range such as 100 to 105.

As a further alternative, the administrator can specify AppleTalk networks from within a partial cable range. For example, an access list statement can target AppleTalk networks 201 to 204 from within the complete cable range of 200 to 205.

As with other access lists, an implicit deny performs the last test of the access list. In AppleTalk, the default is to deny all other network access.

The **access-group** command is used to apply the AppleTalk access lists to one or more interfaces.

By filtering networks, the administrator can permit or prevent data packets and routing update packets on the specified interface. Routing updates use the AppleTalk Layer 3 protocol Routing Table Maintenance Protocol—RTMP.

Access list control involving zone updates controls ZIP traffic. These access lists focus on GetZoneList (GZL) packets. The administrator must use separate access list statements for zone filtering. The procedures involving GZL are covered in the ACRC course.

AppleTalk Access List Commands

Router (config) #

access-list *number* { permit | deny } cable-range *cable-range*

- **Defines full cable-range filtering parameter**

access-list *number* { permit | deny } within cable range *cable-range*

- **Defines partial cable-range filtering parameter**

access-list *number* { permit | deny } other-access

- **Defines default action for other cable ranges**

Router (config-if) #

appletalk access-group *access-list-number*

- **Links traffic filter to an interface**

58

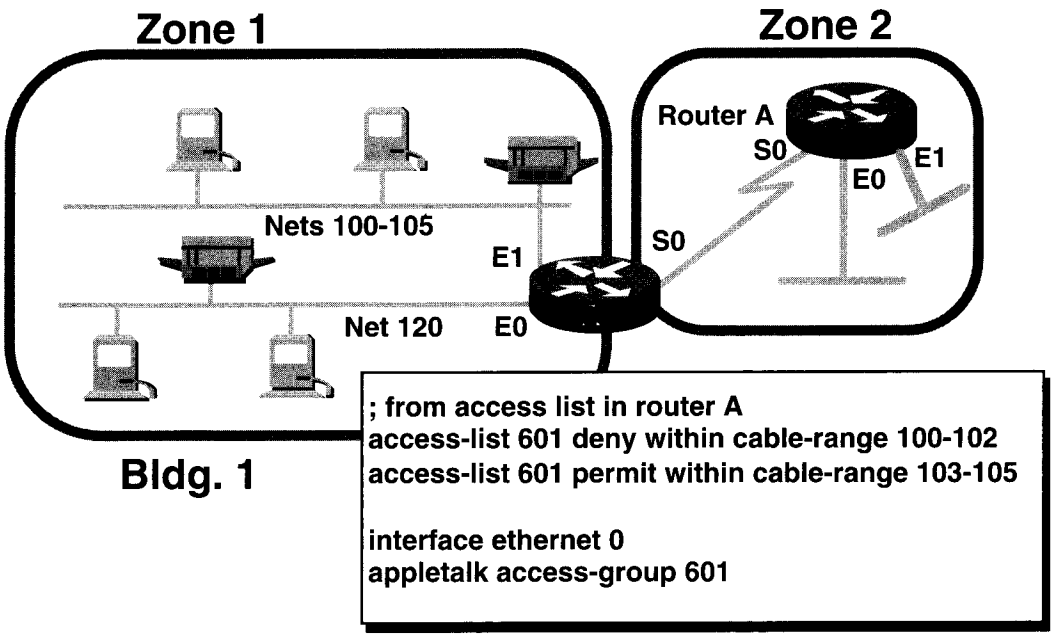
The **access-list** command permits or denies an entire cable range.

The **access-list within cable-range** variation permits or denies part of a cable range. Specify a start and end network number separated by a hyphen.

The **access-list other-access** command defines the default action (permit or deny) to take for other networks or cable ranges.

The **appletalk access-group** command links the access list to one or more specified interfaces.

Controlling Access Example



59

In the example, the access list shows configuration statements in router A:

access-list 601 deny within cable-range 100-102 field descriptions:

601—Specifies this as an AppleTalk access list.

deny—Traffic matching specified parameters will be blocked.

within cable-range 100-102—Sets the range of networks for denial.

permit—Traffic matching specified parameters will be allowed access.

within cable-range 103-105—Set the remaining networks in the cable range for permit.

appletalk access-group 601—Applies list 601 to interface E0 as a cable-range filter for AppleTalk networks.

ZIP Reply Filter Configuration

- **Limit ZIP traffic between routers**

Router (config) #

access-list *number* { permit | deny } zone *zone-name*

- **Defines filtering for specified zone**

access-list *number* { permit | deny } additional-zones

- **Defines default filtering for all other zones**

Router (config-if) #

appletalk zip-reply-filter *access-list-number*

- **Links traffic filter to an interface**

60

Use the **access-list zone** command to create an entry in the zone filter list. It must use an access list number in the number range 600 to 699.

access-list zone Command	**Description**
access-list-number	The number of the access list; an integer in the range 600 to 699.
zone-name	The name assigned to the zone being filtered.

Use the **appletalk zip-reply-filter** command to assign the access list to an incoming interface.

The zip-reply filter limits the zones that are visible from the router by other AppleTalk routers.

appletalk zip-reply-filter Command	**Description**
access-list-number	The number of the access list; an integer in the range 600 to 699.

Monitoring AppleTalk Access Lists

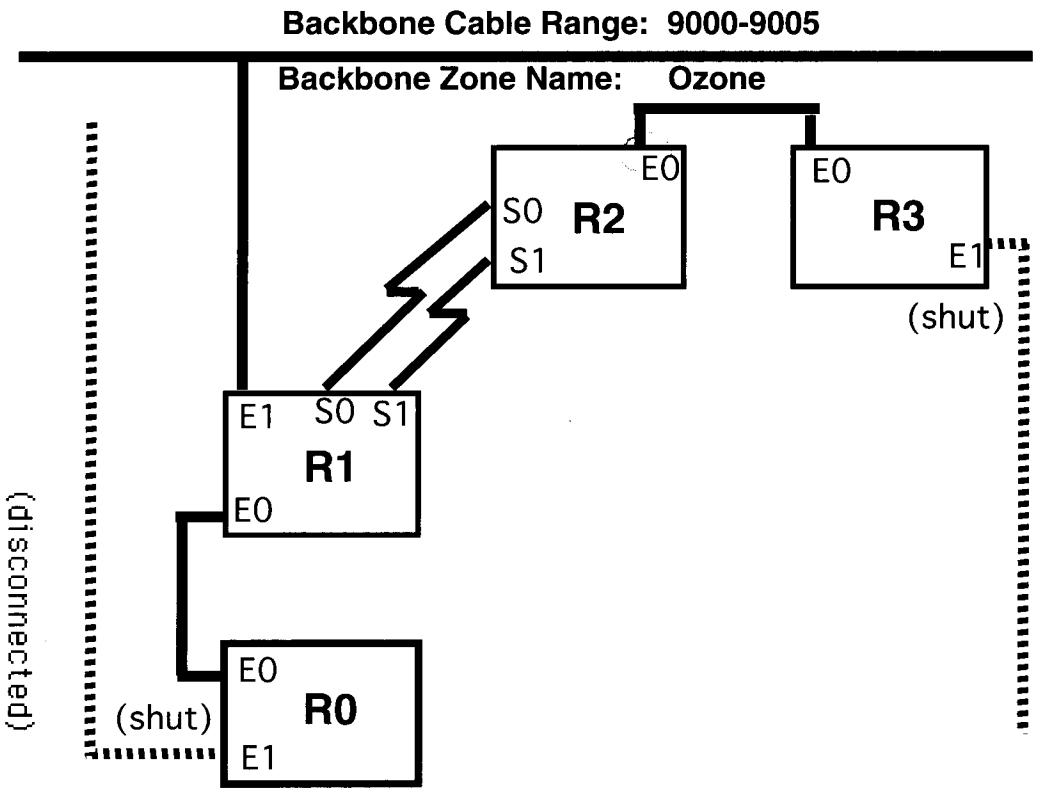
```
Router> show appletalk access-lists

AppleTalk access list 601:
  permit zone ZoneA
  permit zone ZoneB
  deny additional-zones
  permit network 55
  permit network 500
  permit cable-range 900-950
  deny includes 970-990
  permit within 991-995
  deny other-access
```

61

Use the **show appletalk access-lists** command to display the access lists that are set up for AppleTalk.

Lab: AppleTalk Access Lists

AppleTalk Access Lists Data Sheet



Objective: Configure cable-range access lists to control basic AppleTalk traffic.

Step 1 Refer to the AppleTalk Planning Worksheet that contains the AppleTalk DDP cable-range addresses and zone names you used when you configured AppleTalk for your group. Write the router interface cable ranges and zone names onto the graphic or onto the table below:

Routers/Interface	Cable Range	Zone Name
R0/E0—R1/E0	6100 - 6199	ZONE 61
R1/S0—R2/S0	6200 - 6299	ZONE 62
R1/S1—R2/S1	6300 - 6399	ZONE 63
R2/E0—R3/E0	6400 - 6499	ZONE 64

Step 2 Use extended **ping** for AppleTalk to test connectivity between R0 and R3, and between R3 and R0. CLK. 168

Step 3 Shut the E1 connections on R0 and R3, but leave the no shut for the R1 to the backbone.

AppleTalk Access List Implementation

Instructions: Refer to the diagram and information on the AppleTalk Access Lists Data Sheet. Use access lists to prevent packets from flowing between R0 and R3, between R1 and R2, and between R3 and R0. Perform the task appropriate for your router and consult with others in your group about their tasks.

Step 1 Enter AppleTalk access list statements to prevent packets from the cable range out router R0 from accessing router R3. Filter the full cable range. Permit all other AppleTalk packet access. Access-list 600 deny cable-range 6100-6199

Check the statements with your group to reach consensus. Then, if appropriate, enter the agreed-upon access list statements into your router configuration.

Step 2 Enter AppleTalk access list statements to prevent packets from the cable range out router R3 from accessing router R0. Filter the full cable range. Permit all other AppleTalk packet access.

Check the statements with your group to reach consensus. Then, if appropriate, enter the agreed-upon access list statements into your router configuration.

Step 3 Enter AppleTalk access list statements to prevent packets from the cable range out router R1 from accessing router R2. Filter the full cable range. Permit all other AppleTalk packet access.

Check the statements with your group to reach consensus. Then, if appropriate, enter the agreed-upon access list statements into your router configuration.

Step 4 Enter AppleTalk access list statements to prevent packets from the cable range out router R2 from accessing router R1. Filter the full cable range. Permit all other AppleTalk packet access.

Check the statements with your group to reach consensus. Then, if appropriate, enter the agreed-upon access list statements into your router configuration.

Step 5 Use the **show appletalk access-lists** command and **show appletalk interface** command to verify the setting for your access list.

Step 6 Use the extended **ping** command and select the **appletalk** protocol to test access within your group.

#1

access-list 600 deny cable-range 6100-6199
access-list 600 permit other-access

R1 s0
appletalk access-group 600

#

access-list 601 deny cable-range 6400-6499
access-list 601 permit other-access

R1 s0
appletalk access-group 601

Summary

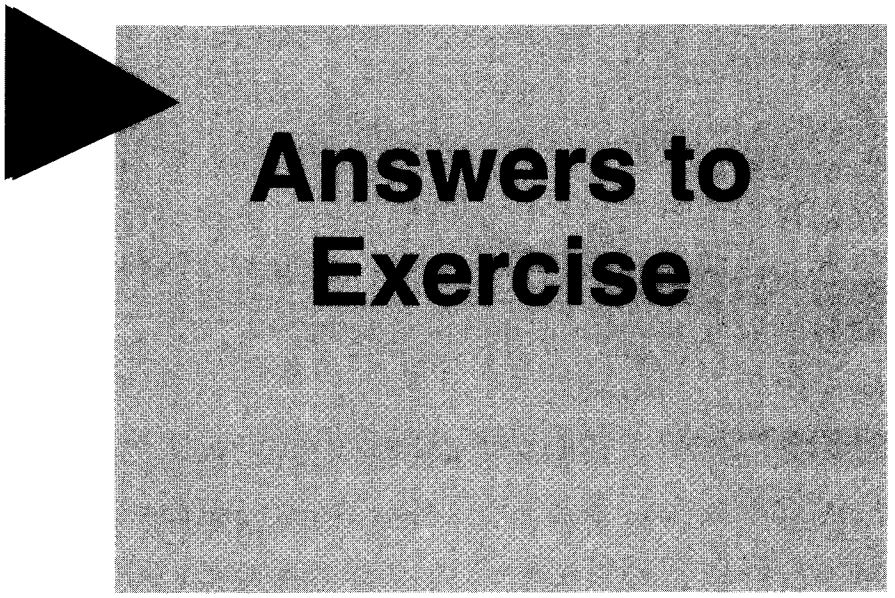
Access lists perform several functions within a Cisco router, including:

- **Implement security/access procedures**
- **Determine whether packets need dialer for WAN links**
- **Act as a protocol "firewall"**

Extended access lists allow filtering on address, protocol, and application parameters

Use access lists to limit broadcast traffic from protocol overhead packets

64



65

Answers to Exercise

Exercise: Novell IPX SAP Filters Configuration

Problem 1

Configuration statements for router R3:

access-list 1001 deny 900.1234.0011.def1 7

access-list 1001 permit -1

int e0

ipx input-sap-filter 1001

Problem 2

Configuration statements for router R1:

access-list 1002 deny 9e.0800.1213.a478 4

access-list 1002 permit -1

int s0

ipx-output-sap-filter 1002

int s1

ipx-output-sap-filter 1002