

Chapter 1

Routing AppleTalk

1

This chapter describes the routing process of the AppleTalk network protocol. The topics and tasks described in this chapter include:

- An overview of the AppleTalk routing protocol.
- Cisco's implementation of AppleTalk on both extended (also known as Phase II) and nonextended (Phase I) interfaces.
- Configuring AppleTalk routing.
- Configuring AppleTalk access list filters.
- Monitoring and debugging an AppleTalk network.

For more detailed information about the AppleTalk network systems, refer to the appendix "References and Recommended Reading."

Cisco's Implementation of AppleTalk

AppleTalk was designed as a client-server, or *distributed* network system. In other words, users share network resources, such as files and printers, with other users. Interactions with servers are essentially transparent to the user, as the computer itself determines the location of the requested material, and accesses it without requesting information from the user.

AppleTalk identifies several network entities, of which the most elemental is a *node*. A node is simply any device connected to an AppleTalk network. The most common nodes are Macintosh computers and laser printers, but many other types of computers are also capable of AppleTalk communication, including IBM PCs, DEC VAX systems and a variety of workstations. A router is considered a node on each connected network. To avoid confusion, these router nodes are referred to as *ports*. The next entity defined by AppleTalk is a *network*. An AppleTalk network is simply a single logical cable. Finally, an AppleTalk *zone* is a logical group of one or more (possibly noncontiguous) networks. These AppleTalk entities are shown in Figure 1-1.

Apple Computer has produced a variety of internetworking products with which to connect AppleTalk local area networks. Apple supports Ethernet, Token Ring, FDDITalk, and its own proprietary twisted-pair media access system (called LocalTalk). However, to allow an AppleTalk network full participation in a multiprotocol internet, a multiprotocol router is required.

All routers from Cisco Systems support the AppleTalk network protocol (both extended and nonextended) over FDDI, Ethernet, Token Ring, synchronous serial, and X.25 interfaces.

Figure 1-1 AppleTalk Entities

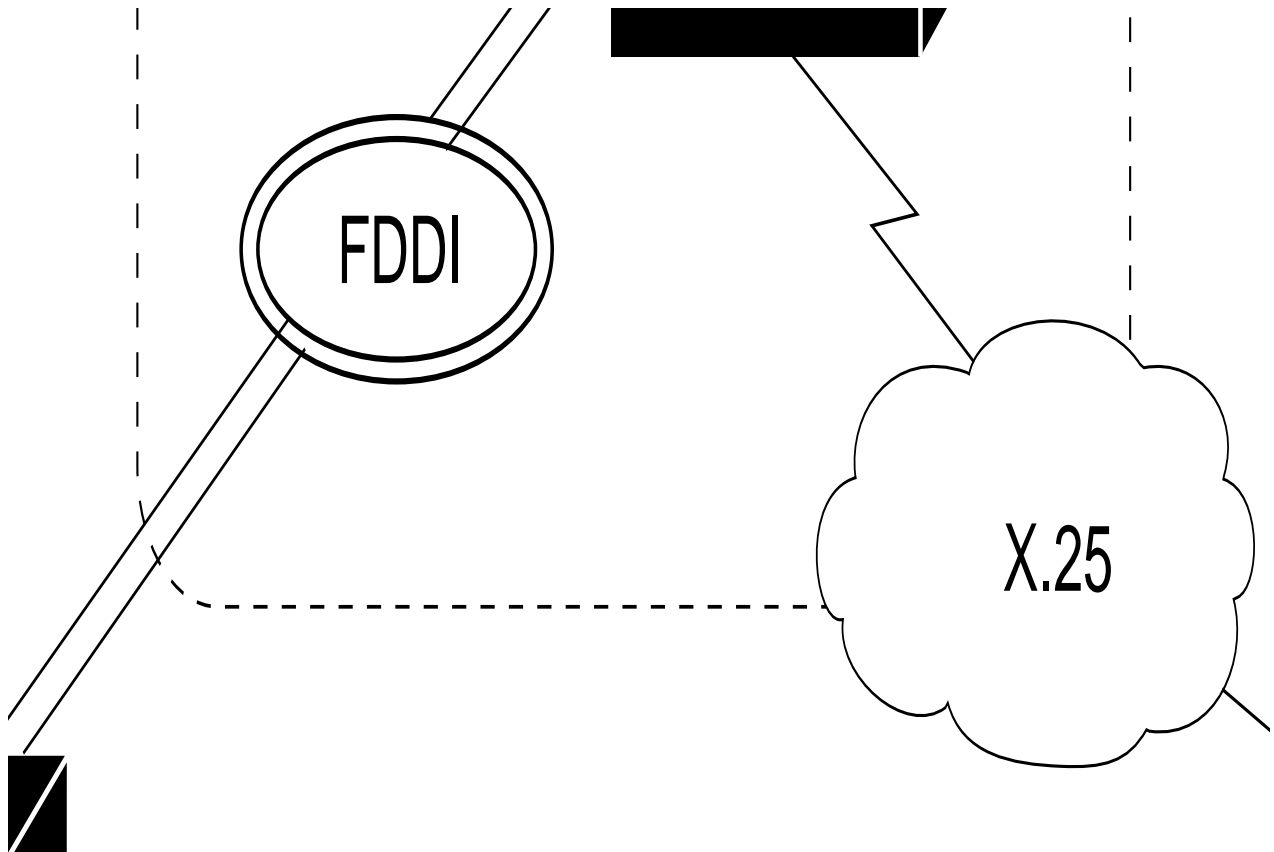


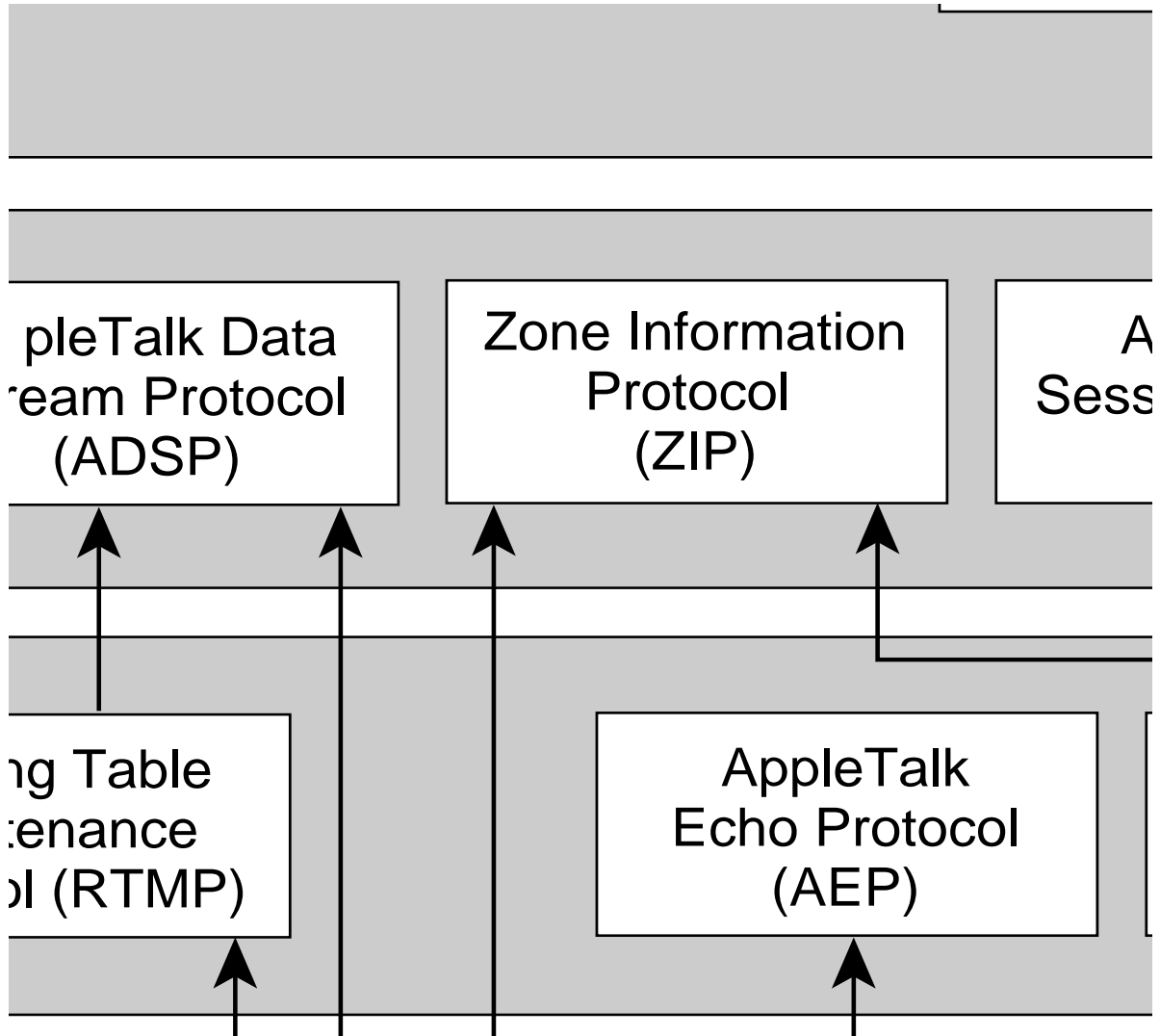
Figure 1-2 compares the AppleTalk protocols with the standard seven-layer OSI model, and illustrates how AppleTalk works with a variety of physical and link access mechanisms.

The Cisco AppleTalk implementation provides the following services:

- AppleTalk Address Resolution Protocol (AARP)
- Datagram Delivery Protocol (DDP)
- Routing Table Maintenance Protocol (RTMP)
- Name Binding Protocol (NBP)
- AppleTalk Echo Protocol (AEP)
- AppleTalk Transaction Protocol (ATP)
- Zone Information Protocol (ZIP)
- AppleTalk Filing Protocol (AFP)
- Print Access Protocol (PAP)

The DDP, RTMP, and AEP protocols provide end-to-end connectivity between internet-worked nodes. NBP maps network names to AppleTalk internet addresses. NBP relies on ZIP to help determine which networks belong to which zones. File and print access is provided through AFP and PAP respectively, which work in concert with applications such as AppleShare and print servers.

Figure 1-2 AppleTalk and the OSI Reference Model



Note: Apple Computer uses the name “AppleTalk” to refer the Apple Networking Architecture, whereas the actual transmission media used in AppleTalk Network are referred to as LocalTalk (Apple Computer’s proprietary twisted pair transmission medium for AppleTalk), TokenTalk (AppleTalk over Token Ring), EtherTalk (AppleTalk over Ethernet), and FDDITalk (AppleTalk over Fiber Distributed Data Interface).

AppleTalk, like many network protocols, makes no provisions for network security. The design of the AppleTalk protocol architecture requires that security measures be executed at higher application levels. Cisco Systems supports AppleTalk distribution lists, allowing control of routing updates on a per interface basis. It is a security feature similar to those provided for other protocols.

Extended (Phase II) Versus Nonextended (Phase I) AppleTalk

AppleTalk was designed for local work groups. With the installation of over 1.5 million Macintoshes in the first five years of the product's life, Apple found that some large corporations were exceeding the design limits of AppleTalk, so they created extended AppleTalk. The extended AppleTalk architecture increases the number of nodes per AppleTalk Internet to over 16 million and an unlimited number of zones per cable. Apple also enhanced AppleTalk's routing capabilities and reduced the amount of network traffic generated by AppleTalk routers.

The introduction of the extended AppleTalk architecture also introduces the concept of *non-extended* and *extended* networks. Nonextended AppleTalk networks are sometimes called "Phase I" and extended networks are called "Phase II." Nonextended networks refer to the nonextended AppleTalk Ethernet 1.0 networks, (explicitly removed by Apple but still supported by Cisco), and to the nonextended serial line-based networks, including those configured using X.25 and LocalTalk.

Extended networks refer to the extended AppleTalk compliant networks configured on Ethernet (EtherTalk 2.0), FDDI, and Token Ring media. Samples of the AppleTalk nonextended and extended network configurations can be found in the section "AppleTalk Configuration Examples."

The AppleTalk extended-network architecture provides extensions compatible with nonextended AppleTalk internets. The AppleTalk extended architecture was designed to remove the previous limits of 254 concurrently active AppleTalk nodes per cable, as well as the previous limit of one AppleTalk zone name per cable. Extended AppleTalk contains better algorithms for choosing the best routers for traffic and is designed to minimize the amount of broadcast traffic generated for routing updates.

Another important feature in extended AppleTalk is the ability of a single AppleTalk cable to be assigned more than one network number. The size of the range of network numbers assigned to a cable determines the maximum number of concurrently active AppleTalk devices that can be supported on that cable, which is 254 devices per network number.

Cisco routers running software Release 8.2 or later support both extended and nonextended AppleTalk. An interface may be configured for either extended or nonextended AppleTalk operation. This allows an easy transition between the protocols; however, see the guideline in the section "Configuration Guidelines" before mixing versions of AppleTalk in your network.

Nonextended AppleTalk Addressing

AppleTalk addresses are 24 bits long. They consist of two components: a 16-bit network number, and an 8-bit node number. The Cisco AppleTalk software parses and displays these addresses as a sequence of two *decimal* numbers, first the network number, then the node number, separated by a dot. For example, node 45 on network 3 is written as 3.45. A node is any AppleTalk-speaking device attached to the network. Each enabled AppleTalk interface on a router is a node on its connected network.

AppleTalk Zones

When a router is used to join two or more AppleTalk networks into an internet, the component physical networks remain independent of each other. A network manager may assign to these network conceptual groupings known as *zones*.

There are two main reasons to create zones in an AppleTalk internet: to simplify the process of locating and selecting network devices, and to allow for the creation of departmental work groups which may exist on several different and possibly geographically separated networks.

For example, consider a large AppleTalk internet which may contain hundreds or thousands of shared resources and devices. Without a method of dividing this large number of resources and devices into smaller groups of devices, a user might have to scroll through hundreds or thousands of resource/device names in the Chooser to select the one resource to be used. By creating small, conceptual groups of resource and device names, a user may now choose the resource they need much more quickly and easily than if they were sorting through a very long list of names.

A zone may include many networks, which need not be physically co-located. A zone is not limited by geographical area. The partitioning afforded by zone names is conceptual, not physical.

Zones are defined by the network manager during router configuration. When a Cisco router is configured, each AppleTalk-configured interface must be associated with exactly one zone name for nonextended networks, or one or more zone names for extended networks. Until a zone name has been assigned, AppleTalk routing features are disabled for that interface. The section “Configuring AppleTalk Routing” later in this chapter, describes the subcommands to use in the zone naming process.

It is very important that routers explicitly configured with zone information be configured correctly.

Name Binding Protocol (NBP)

The Name Binding Protocol (NBP) maps network entity names with internet addresses. It allows users to specify descriptive or symbolic names, while other software processes refer to numerical addresses for the same entities. With NBP, almost all user-level programs respond to names instead of numbers. When users select an AppleTalk device, they are using the NBP protocol to bind the address to the device name. Numerical addresses assigned to zones are primarily used by the router software and by network managers in the **ping** process (see the section “The AppleTalk Ping Command” later in this chapter).

NBP provides four basic services for binding names to nodes and zones:

- Name registration
- Name deletion
- Name look-up
- Name confirmation

The nature of the AppleTalk addressing scheme is inherently volatile and node addresses change frequently. Therefore, NBP associates numerical addresses with aliases which continue to reference the correct address if the address changes.

Zone Information Protocol (ZIP)

NBP uses the Zone Information Protocol (ZIP) to determine which networks belong to which zones. A Cisco router uses ZIP to maintain the network-number-to-zone-name mapping of the AppleTalk internet.

Along with a routing table, each router maintains a data structure known as the *zone information table* (ZIT). The table provides a listing of network numbers for each network in every zone. Each entry is a *tuple* (an inseparable network-number-hop-number set) that matches a network number with a zone name as supplied by the network manager.

Dynamic Address Assignment

AppleTalk provides for *dynamic address assignment*. With dynamic address assignment, not all fields of an AppleTalk address need to be specified in the configuration of a router. If there is another AppleTalk router on the network, it may be able to supply the network number. A preconfigured router on an AppleTalk network acts as a *seed router*, sending out address information to other routers on its connected network.

Seed routers are routers that come up and verify the configuration. If the configuration is valid, they start functioning. Seed routers come up even if no other routers are on the network. On the other hand, *nonseed routers* must first communicate with a seed router before it can function. A nonseed router must verify the configuration with another functioning router. The configuration must match exactly for the router to function.

A node number can be chosen by a network manager, but, alternatively, may be negotiated between AppleTalk hosts on the network. Note that network numbers may be assigned by routers only.

Unspecified parts of the AppleTalk address are entered as zero. For example:

34.5	Represents a fully qualified address (net 34, node 5).
0.5	Is a partially qualified address (net unspecified, node 5).
122.0	Represent net 122, node unspecified.
0.0	Is completely unspecified.

Node numbers are automatically assigned by AppleTalk (in other words, configured as zero)

or when the specified address is in use it randomly chooses an initial value. The node will first try the node number that was its most recent address. If that number is unavailable, the node then searches for the next available address. If it reaches 254 without finding an available number, it cycles back to 1 and continues until it finds a free address. Nonextended network address restrictions are as follows: user node numbers are from 1 to 127 while servers/printer node numbers are from 128 to 254.

For nonseed routers, an interface will participate in the routing of only local traffic until its network number has been determined. If zero has been specified for a network number, that interface will not route any packets until it receives its network number from a seed router.

Receipt of a routing table update informs the router of the network number for the interface on which the packet was received. Every routing table update includes the network number of the network the packet was sent on. Therefore, the router is able to determine the network number of the receiving interface.

As long as one fully configured (seed) router exists on an Ethernet or Token Ring cables, other interfaces and routers directly attached to that cable need not be configured; they can take their information from the initial router. However, once the configuration process has stabilized for a particular AppleTalk internet, all routers thereafter should be configured as seed routers. Note that synchronous serial and X.25 network interfaces must be explicitly configured on each router to be used as AppleTalk transports.

Note: The same Macintosh may have several different AppleTalk addresses over time. Routing tables, maintained by RTMP, contain network and zone information.

RTMP routing tables contain an entry for each network a datagram can reach. Each entry includes the router port which leads to the destination network, the node ID of the next router to receive the packet, and the distance in hops to the destination network. Periodic exchange of routing tables allows the routers in an internet to ensure that they supply current and consistent information.

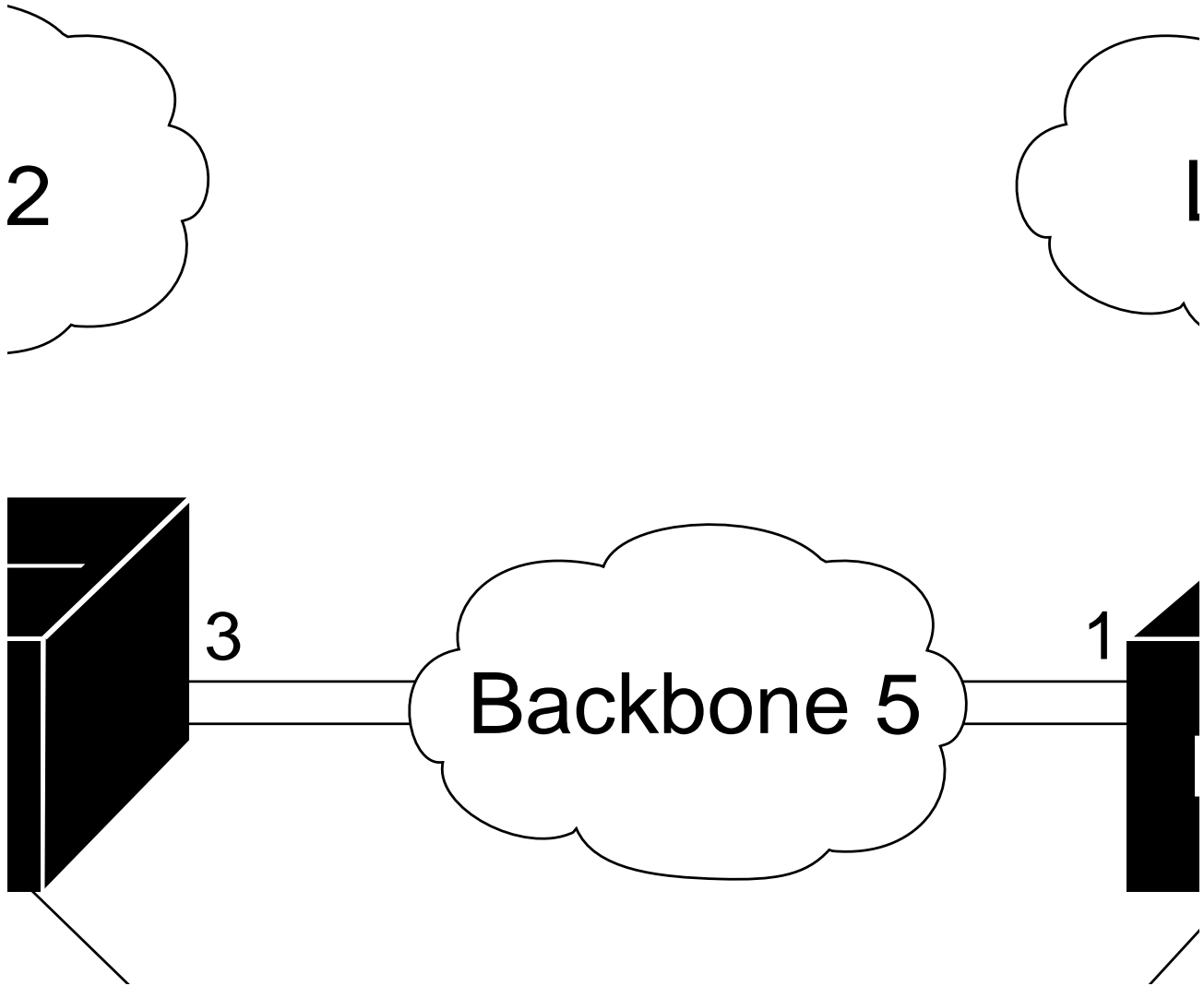
Node information is maintained by tables appropriate to the media (usually AARP tables).

Figure 1-3 shows a sample RTMP table and the corresponding network topology.

Extended AppleTalk Addressing

AppleTalk addresses, as explained in the section “Nonextended AppleTalk Addressing,” earlier in this chapter, are composed of a 16-bit network and an 8-bit node number. In nonextended AppleTalk, nodes within a single cable can communicate using only their 8-bit node numbers.

Figure 1-3 Sample AppleTalk Routing Table



A node in extended AppleTalk is *always* identified by its network *and* node number. Dynamic address resolution when a router is not present includes the assignment of a random network number within a small range, as well as a node number. When a router is present in the network, a node starts up using its newly acquired address for a short period of time. It then immediately requests a list of valid network numbers from the router or routers. The node then uses these to determine its actual AppleTalk address by selecting an unassigned address. (Be aware that this two-step process causes the nonextended AppleTalk to be incompatible with the extended AppleTalk routing.)

A new concept of cable *ranges* is introduced with the extended AppleTalk. Cables now have ranges of network numbers and multiple zones that may exist on them, so that a node can access anything that is in any of the zones that are on the same cable as the node itself. But the node can exist in only one zone and on only one network.

In an extended AppleTalk network, the mapping of a single network number to a zone name is no longer valid. End nodes are expected to know the zone to which they belong, or to choose from the list of available zones provided by a router. The router maintains a default zone which new nodes will use automatically if they have not previously chosen a zone.

Configuring AppleTalk Routing

This section provides an overview on how to configure Cisco AppleTalk routing.

Configuration Overview

The AppleTalk interface configuration is different for the two types of AppleTalk interfaces; extended and nonextended.

Configuring a nonextended AppleTalk interface involves the following steps:

- Step 1:** Enable AppleTalk routing with the **appletalk routing** command.
- Step 2:** Assign the nonextended AppleTalk addresses with the **appletalk address** interface subcommand.
- Step 3:** Assign the zone name with the **appletalk zone** interface subcommand.

Configuring an extended AppleTalk interface involves these steps:

- Step 1:** Enable AppleTalk routing with the **appletalk routing** command.
- Step 2:** Assign the extended AppleTalk cable range parameters with the **appletalk cable-range** command.
- Step 3:** Assign the zone name or names with the **appletalk zone** interface subcommand.

The software also provides commands for fine tuning the AppleTalk network, for configuring packet filtering mechanisms, monitoring, maintaining and troubleshooting network operation. Alphabetically arranged summaries of the commands described in this chapter are also provided at the end of the chapter.

Configuration Guidelines

Follow these guidelines when configuring your AppleTalk network on a Cisco router:

- A Macintosh that contains an Ethernet card must run EtherTalk version 2.0 or later to support extended AppleTalk. A Macintosh with only a LocalTalk interface does not require any changes.
- Shiva FastPath routers must run K-Star version 8.0 or later and be explicitly configured for extended AppleTalk.
- Apple's Internet Router software, version 2.0, supports a transition mode for translation between the nonextended AppleTalk and the extended AppleTalk on the same network.

- For the Internet Router version 1.0, transition mode requires the Apple upgrade utility and a special patch file from Apple.
- If your AppleTalk internet contains any routers that support only nonextended AppleTalk, the following configuration restrictions must be observed. These restrictions are not enforced, but unpredictable behavior may result if they are violated. All routers in a network must support extended AppleTalk before these restrictions may be lifted.
 - Cable ranges of only one (666-666, for example) are permitted.
 - Each AppleTalk network may have only one zone associated with it.

A general understanding of Cisco's representation of AppleTalk addresses is necessary before configuration of the router. Refer to the sections "Cisco's Implementation of AppleTalk," "Nonextended AppleTalk Addressing," and "Extended AppleTalk Addressing" earlier in this chapter.

Enabling AppleTalk Routing

Before you can configure AppleTalk routing, you need to enable AppleTalk protocol processing. To do that, use the **appletalk routing** global configuration command. The full command syntax follows:

```
appletalk routing  
no appletalk routing
```

The **appletalk routing** configuration command enables AppleTalk protocol processing. The **no appletalk routing** disables all AppleTalk processing.

Assigning Nonextended (Phase I) AppleTalk Address

To assign AppleTalk addresses for nonextended networks, use the **appletalk address** interface subcommand. Its full syntax follows.

```
appletalk address address  
no appletalk address
```

The argument *address* assigns AppleTalk addresses on the interfaces that will be used for the AppleTalk protocol. It assigns one AppleTalk address per interface. This step must be done before assigning zone names.

Note: Use this subcommand to configure nonextended interfaces.

The **no appletalk address** subcommand disables nonextended AppleTalk processing on the interface.

Example:

These commands begin AppleTalk routing and assign address 1.129 to interface Ethernet 0.

```
appletalk routing
!
interface ethernet 0
appletalk address 1.129
```

Assigning a Cable Range for Extended AppleTalk (Phase II)

To assign the cable-range parameters, use the **appletalk cable-range** interface subcommand. The full command syntax follows.

```
appletalk cable-range start-end [network.node]
no appletalk cable-range
```

This command designates an interface to be on an extended AppleTalk network. A cable range is the network numbers assigned to an extended network.

This range is specified using the argument *start-end*, which is a pair of decimal numbers between 1 and 65,279, inclusive. The starting network number should be less than or equal to the ending network number.

Note: To use this subcommand to configure a nonextended AppleTalk interface, the starting and ending network numbers must be equal.

Specifying a cable range of 0-0 in the *start-end* argument (start = end = 0) places the interface into discovery mode, which attempts to determine cable range information from another router on that network.

The optional *network.node* argument specifies the suggested network and node number that will be used first when selecting the AppleTalk address for this interface. Note that any suggested network number must fall within the specified range of network numbers.

Use the **no appletalk cable-range** command to disable AppleTalk processing on the interface.

Example:

This command assigns a cable range of 2-2 to the interface:

```
appletalk cable-range 2-2
```

Assigning a Zone Name

Use the **appletalk zone** interface subcommand to assign a zone name to an AppleTalk in-

terface. Full command syntax for this command follows:

```
appletalk zone zonename  
no appletalk zone [zonename]
```

Interfaces which are configured for seed routing or which have discovery mode disabled must have a zone name assigned before AppleTalk processing will begin.

The argument *zonename* specifies the name of the zone for the connected AppleTalk network. The argument *zonename* may include special characters from the Apple Macintosh character set. To include a special character, insert a colon and two uppercase hexadecimal characters. The hexadecimal equivalent for special characters in the Macintosh character set may be found in character tables published by Apple Computer (see Appendix D in the text *Inside AppleTalk*, 2nd edition).

The **appletalk zone** command works for the nonextended AppleTalk interface, however, you may repeat the command to define multiple zones in an extended AppleTalk network only when a cable range has been specified.

The router selects the zone in which it will operate from the list specified. The first zone specified in the list is the *default zone*. Computers in the network will select the zone in which they will operate from the list of zone names valid on the cable to which they are connected. If an interface is using nonextended AppleTalk, repeated execution of the zone command will replace the zone name for the interface with the newly specified zone name.

The **no appletalk zone** interface subcommand deletes a zone name from a zone list or the entire zone list if none is specified. The optional zone name is ignored for nonextended AppleTalk interface configurations. The command is also ignored if the specified zone name is not in the current zone list for an interface. The list should be cleared using the **no appletalk zone** interface subcommand before configuring a new zone list.

Note: The zone list is cleared automatically when **appletalk address** or **appletalk cable-range** commands are used.

Examples:

This command assigns zone name Twilight to the interface:

```
appletalk zone Twilight
```

The following example shows use of the AppleTalk special characters sets by setting the zone name to *cisco*zone*.

```
appletalk zone cisco:A5zone
```

Setting and Resetting Discovery Mode

The discovery mode is set and reset using the **appletalk discovery** interface subcommand. The full syntax of this command follows:

appletalk discovery
no appletalk discovery

This command resets the discovery mode and allows a new cable range to be discovered. If the port information has been discovered, and the port is operational, then this command results in the port being a valid seed port.

Use the **no appletalk discovery** command to return the software to the default (off) state.

Configuring IP Encapsulation of AppleTalk Packets

Use the **appletalk iptalk** interface subcommand to encapsulate AppleTalk in IP packets in a manner compatible with the Columbia AppleTalk Package (CAP) IPtalk and the Kinetics IPtalk (KIP) implementations.

appletalk iptalk *net.node zone*

This command enables IPtalk encapsulation on an interface which already has an configured IP address. The command allows AppleTalk communication with UNIX™ hosts running older versions of CAP (Columbia AppleTalk Package) which do not support native AppleTalk EtherTalk encapsulations. Typically, Apple Macintosh users wishing to communicate with these servers would have their connections routed through a Kinetics FastPath™ router running KIP (Kinetics IP) software.

This command is provided as a migration command; newer versions of CAP provide native AppleTalk EtherTalk encapsulations and the IPtalk encapsulation is no longer required. The Cisco implementation of IPtalk assumes that AppleTalk is already being routed on the backbone, since there is currently no LocalTalk hardware interface for Cisco routers.

The Cisco implementation of IPtalk does not support manually configured AppleTalk-to-IP address mapping (**atab**). The address mapping provided is the same as the Kinetics IPtalk implementation when the **atab** facility is not enabled. This address mapping functions as follows: The IP subnet mask used on the router Ethernet interface on which IPtalk is enabled is inverted (one's complement). This result is then masked against 255 (0xFF hexadecimal). This is then masked against the low-order 8 bits of the IP address to obtain the AppleTalk node number. The following example configuration should make this more clear.

Example:

```
interface Ethernet 0
ip address 131.108.1.118 255.255.255.0
appletalk address 20.129
appletalk zone Native AppleTalk
appletalk iptalk 30.0 UDPZone
```

In this configuration, the IP subnet mask would be inverted:

```
255.255.255.0 inverted yields: 0.0.0.255
```

Masked with 255 it yields 255, and masked with the low-order 8 bits of the interface IP address it yields 118.

This means that the AppleTalk address of the Ethernet 0 interface seen in the UDPZone zone is 30.118. This caveat should be noted, however: Should the host field of an IP subnet mask for an interface be more than 8 bits wide, it will be possible to obtain conflicting AppleTalk node numbers. For instance, consider a situation where the subnet mask for the Ethernet 0 interface above is 255.255.240.0, meaning that the host field is 12 bits wide.

Configuring IP Encapsulation DDP Socket to UDP Port Mapping

Use the global configuration command **appletalk iptalk-baseport** to specify the UDP port number which is the beginning of the range of UDP ports used in mapping AppleTalk *well-known* DDP socket numbers to UDP ports. The command syntax looks like this:

```
appletalk iptalk-baseport port-number
```

Implementations of IPTalk prior to April, 1988 mapped well-known DDP socket numbers to privileged UDP ports start at port number 768. In April of 1988, the NIC assigned a range of UDP ports for the defined DDP well-known sockets starting at UDP port number 200 and assigned these ports the names *at-nbp*, *at-rtmp*, *at-echo* and *at-zis*. The Columbia AppleTalk Package, Release 6 and later, dynamically decides which port mapping to use. If there are no AppleTalk service entries in the */etc/services* file, CAP will use the old 768-based mapping.

This is the default UDP port mapping supported by Cisco's implementation of IPTalk. If there are service entries in the */etc/services* file for the AppleTalk services, the Cisco router configured for IPTalk encapsulation should specify the beginning of the port mapping range with the **appletalk iptalk-baseport** command. The following example configuration builds upon the example for the **appletalk iptalk** command to illustrate this concept.

Example:

```
interface Ethernet 0
ip address 131.108.1.118 255.255.255.0
appletalk address 20.129
appletalk zone Native AppleTalk
appletalk iptalk 30.0 UDPZone
appletalk iptalk-baseport 200
```

Checking Packet Routing Validity

Use the **appletalk strict-rtmp** global configuration command to enforce maximum checking of routing packets to ensure their validity. The full command syntax follows:

```
appletalk strict-rtmp
no appletalk strict-rtmp
```

The default of this command is to provides maximum checking.

Currently, strict RTMP checking consists of discarding RTMP arriving from routers not directly connected to the router performing the check. (In other words, no routed RTMP packets will be accepted.)

Use the **no appletalk strict-rtmp** command to disable the maximum checking mode.

Enabling and Disabling Routing Updates

Use the global configuration command **appletalk send-rtmp** to allow the transmission of routing updates to be disabled. The full syntax of the command is:

```
appletalk send-rtmp  
no appletalk send-rtmp
```

This command allows a router to be placed on a network with AppleTalk routing enabled, but without being seen by other AppleTalk routers on the cable. The default is to send routing updates. The **no appletalk send-rtmp** command disables this default.

Changing Routing Timers

Use the global configuration command **appletalk timers** to change the time intervals used in AppleTalk routing, as follows:

```
appletalk timers update-interval valid-interval invalid-interval
```

The argument *update-interval* is the time, in seconds, between routing updates sent to other routers on the network. This is ten seconds by default.

The argument *valid-interval* is amount of time, in seconds, that the router will consider a route valid without having heard a routing update for that route. This is normally twice the update interval, 20 seconds by default. Once this period of time has elapsed without having heard a routing update for a route, the route becomes “suspect,” indicating that a routing update has been missed.

The argument *invalid-interval* is the amount of time, in seconds, that the router will wait before marking a route invalid. Once a route has been marked invalid, the route will no longer be sent to other routers in routing updates. By default, the *invalid-interval* argument is three times the update-interval, or 30 seconds.

Note: Modification of the routing timers should not be undertaken without fully understanding the ramifications of doing so. Many other AppleTalk router vendors provide no facility for modifying their routing timers; should you adjust a Cisco router’s AppleTalk timers such that routing updates do not arrive at these other routers within the normal interval, it is possible to degrade or destroy AppleTalk network connectivity.

Example:

This command increases the update interval to 20 seconds, the route valid interval to 40 seconds, and the route invalid interval to 60 seconds.

```
appletalk timers 20 40 60
```

Assigning a Proxy Network Number

When an AppleTalk internetwork contains routers which support only nonextended AppleTalk and routers which support only extended AppleTalk, then one **apple proxy-npb** global configuration command is required for each zone in which there is a router which supports only nonextended AppleTalk. The full syntax of this command follows.

```
appletalk proxy-npb network-number zonenumber  
no appletalk proxy-npb network-number zonenumber
```

The argument *network-number* must be a unique network number which will be advertised via this router as if it were a real network.

The argument *zonenumber* is the name of the zone requiring compatibility support.

No router may have the same network number defined as a proxy network, and it cannot be associated with a physical network.

Only one proxy is needed to support a zone, but additional proxies may be defined with different network numbers if redundancy is desired. Each proxy will generate one or more packets for each forward request it receives. All other packets sent to the proxy network are discarded. Redundant proxies increase the NPB traffic linearly.

Assume your network topology looks like the one in Figure 1-4. Also assume that Router A supports only nonextended AppleTalk, that Router B supports only extended AppleTalk (not in transition mode), and that Router C supports only extended AppleTalk.

Figure 1-4 Example Network Topology



If router C generates a NBP hookup request for zone A, router B will convert this request to a forward request and send it to router A. Since router A supports only nonextended AppleTalk, it does not handle the forward request and ignores it. Hence, the NBP lookup from router C fails.

To work around this problem without putting a transition router adjacent to the nonextended only router (router A), you could configure router D with a NBP proxy.

If you configured router D with a NBP proxy as follows, any forward requests received for zone A are converted into lookup requests, and therefore, the nonextended router for Net 60 can properly respond to NBP hookup requests generated beyond router C. The following example demonstrates the command needed to describe this configuration.

Example:

```
appletalk proxy 60 A
```

Generating Checksum Verification

Use the **appletalk checksum** global configuration command to enable the generation and verification of checksums for all AppleTalk packets. The full command syntax follows:

```
appletalk checksum  
no appletalk checksum
```

An incoming packet with a nonzero checksum will be verified against that checksum and discarded if in error. By default, checksum verification is enabled.

Cisco routers no longer check checksum on routed packets, thereby eliminating the need to disable checksum to allow operation of some networking applications.

Use the **no appletalk checksum** command to disable checksum verifications.

Specifying the Time Interval Between AARP Transmissions

Use the **appletalk arp interval** global configuration command to specify the time interval between retransmission of ARP packets, as follows:

```
appletalk arp interval milliseconds
```

The argument *milliseconds* specifies the interval. The default and minimum value is 33 milliseconds.

Lengthening the interval between packets permits responses from certain devices which respond more slowly, such as printers and overloaded file servers, to be received.

Example:

This command lengthens the AARP retry interval to 100 milliseconds.

```
appletalk arp interval 100
```

Specifying the AARP Retransmission Count

Use the **appletalk arp retransmit-count** global configuration command to specify the number of retransmissions that will be done before abandoning address negotiations and using the selected address.

```
appletalk arp retransmit-count count
```

The argument *count* specifies the retransmission count. The minimum value that can be specified is 1; the default is 10.

Example:

This command specifies an AARP retransmit count of 25.

```
appletalk arp retransmit-count 25
```

AppleTalk Access and Distribution Lists

An *access list* is a list of AppleTalk network numbers kept by the Cisco router to control access to or from specific networks for a number of services.

Cisco's AppleTalk access lists provide network security by permitting or denying certain packets onto a network interface. Cisco's AppleTalk access lists are applicable to *networks only*; they may not be used for specific nodes. Network managers who are familiar with Cisco's access list support for other network protocols should note that these lists do not actually hide a network. An access-controlled network will still appear in routing information;

however, the network interface will simply not allow packets to pass through it. Network managers who require a more in-depth security strategy should execute measures within the higher-level protocols, such as the AppleTalk Filing Protocol (AFP).

When defining access lists for an interface, all networks within a zone should be governed by the same access control. Although access lists are applied to network numbers, they should be created with attention to the zone in which they are located. This precaution serves two purposes. First, it prevents the partitioning of zones, which should always be a nondivisible unit of an internet. Second, it allows a router to query all the network interfaces within a zone, which prevents duplicate names among nodes.

To simplify the definition of access lists, some network managers may prefer to give each network its own zone name.

A *distribution list* is a list of AppleTalk access list numbers kept by the Cisco router which controls whether the network numbers specified by the access list are processed during the reception or transmission of routing updates. A distribution list will not prevent packets destined for a specified network number from being accepted; it will only prevent the route to the specified network from appearing in neighboring routers' AppleTalk routing tables.

Assigning an Access List to an Interface

An AppleTalk access list is assigned to an interface with the **appletalk access-group** global configuration command. Once assigned, no packet which fails the **appletalk access-list** command will go out on that interface. The full syntax of this command follows:

```
appletalk access-group list  
no appletalk access-group list
```

The argument *list* specifies the appropriate AppleTalk access list. Use the **no appletalk access-group** command to remove the list from the interface.

The EXEC command **show appletalk traffic** displays the number of packets dropped because of access control. Refer to the section on "Monitoring the AppleTalk Network" later in this chapter for more information. See the section "Filtering Networks Sent Out in Updates" for an example of the use of this command.

Controlling Interface Access

To permit or deny packets onto a specific network interface, use the **access-list** global configuration command, as follows:

```
access-list list {permit | deny} network  
no access-list list
```

The argument *list* is an integer between 600 and 699 and the argument *network* is an AppleTalk network. A *network* argument of -1 represents any network.

Additional permit and deny conditions may be added to the list by issuing further **access-list** commands for that list. Note that the order of specification, especially of the all networks entry, matters in how the access list treats networks.

Use the **no access-list** command to remove an entry from the list.

Example:

These commands will not permit packets destined for network number 101 to transmit the interface Ethernet 6 packets.

```
interface ethernet 6
  appletalk cable-range 71-71 71.0
  appletalk zone Tir'n na n'Og
  access-list 602 deny 101
```

Filtering Networks Received in Updates

Use the **apple distribute-list** interface subcommand to filter input from the networks. The full syntax for this command follows.

```
appletalk distribute-list access-list-number in  
no appletalk distribute-list access-list-number in
```

The argument *access-list-number* is the number of a predefined access list.

Use the keyword **in** to filter networks received in update. AppleTalk network numbers specified by the *access-list-number* argument will not be inserted into the router's AppleTalk routing table when routing updates are received. Use the **no appletalk distribute-list** command to remove this function.

This is a special command providing compatibility between nonextended-only AppleTalk routers (Cisco pre-8.2 release) and the extended AppleTalk routers (Apple's Internet Router, WellFleet, NSC, and so on).

Example:

These commands prevent the insertion of routing updates from network 10 into the routing table.

```
access-list 601 deny 10
!  
interface ethernet 3  
  appletalk distribute-list 601 in
```

Filtering Networks Sent Out in Updates

Use the interface configuration subcommand **apple distribute-list** to filter output from networks. The full syntax of this command follows.

appletalk distribute-list *access-list-number* **out**
no appletalk distribute-list *access-list-number* **out**

The argument *access-list-number* is the number of a predefined access list.

Use the keyword **out** to suppress the AppleTalk networks specified by the *access-list-number* from being sent in updates. Use the **no appletalk distribute-list** command to remove the filter.

This is a special command providing compatibility between the nonextended-only AppleTalk routers (Cisco pre-8.2 release) and the extended AppleTalk routers (Apple's Internet Router, WellFleet, NSC, and so on).

Example:

These commands prevent network 10 on another Ethernet interface from publishing updates on interface Ethernet 0. The **appletalk access-group** command prevents packets from being sent out the interface.

```
access-list 601 deny 10
interface Ethernet 0
appletalk distribute-list 10 out
appletalk access-group 601
```

AppleTalk Configuration Examples

The following examples illustrate configurations of nonextended AppleTalk networks routing between two Ethernets, over HDLC, and X.25—and a configuration for an extended AppleTalk network.

Nonextended AppleTalk Routing Between Two Ethernets

This example configuration illustrates how to configure routing between two Ethernets. Ethernet 0 is on network 1, at node 128. Ethernet 1 is on network 2, at node 154. The two networks are in the “Twilight” and “No Parking” zones, respectively. See Figure 1-5 for an illustration.

Figure 1-5 Nonextended AppleTalk Routing Between Two Ethernet Networks

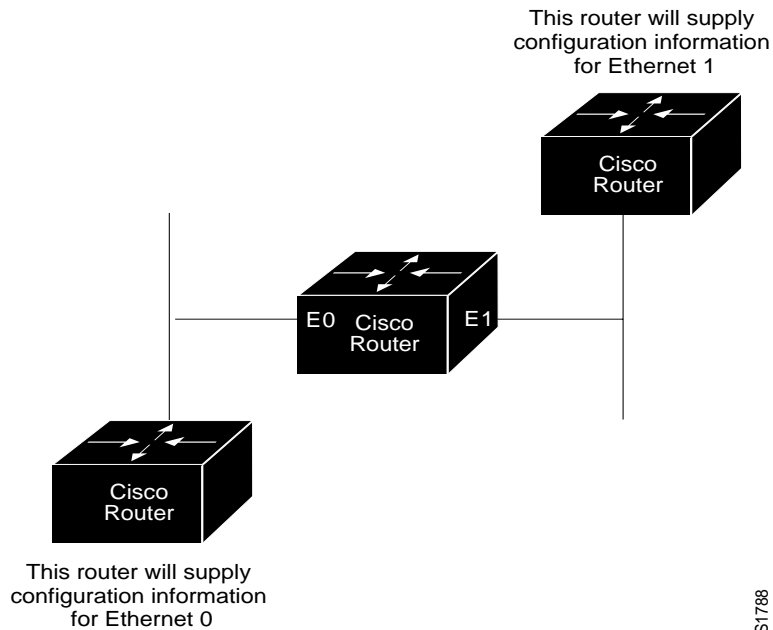


Example 1:

```
appletalk routing
!  
interface ethernet 0  
appletalk address 1.128  
appletalk zone Twilight  
!  
interface ethernet 1  
appletalk address 2.154  
appletalk zone No Parking
```

Example 2 is a variation of the above configuration. It differs in that it has other seed routers on both networks to provide the zone and network number information. In this way, the Cisco router discovers the information dynamically. Refer to Figure 1-6 for an illustration.

Figure 1-6 Routing Between Seed Routers



Example 2:

```

appletalk routing
!
interface ethernet 0
appletalk address 0.0
!
interface ethernet 1
appletalk address 0.0

```

Configuring Transition Mode

The Cisco router may be used to route between extended and nonextended AppleTalk networks that exist on the same cable. Many other vendors have coined the term *transition mode* for this type of routing.

To do this on the Cisco router, you must have two ports connected to the same physical cable. One port will be configured as a nonextended AppleTalk network.

Both ports must have unique network numbers because you are actually routing between two separate AppleTalk networks, an extended and a nonextended network. Figure 1-7 shows an example of the topology and configuration of such connection.

Figure 1-7 Transition Mode Topology and Configuration



Example:

```
interface ethernet 0
  appletalk cable-range 2-2
  appletalk zone No Parking
!
interface ethernet 1
  appletalk address 3.128
  appletalk zone Twilight
!
interface ethernet 2
  appletalk cable-range 4-4
  appletalk zone Do Not Enter
```

Note: Networks 2-2 and 4-4 in the above example have cable range of one and a single zone in their zone list. This must be true to maintain compatibility with the nonextended network, network 3.

Nonextended AppleTalk Routing over HDLC

AppleTalk's dynamic address assignment feature allows users and network managers to choose default network addresses. Configuring AppleTalk nodes over HDLC encapsulation has one major variation from normal configuration procedures: you must explicitly specify all node numbers on each end of a serial line, and at least one end of the line must also be provided with a network number. A router will not respond to the specification of zero for any node number. Moreover, if two nodes are assigned the same node numbers by user configuration, a router in an HDLC environment will not acknowledge any conflict.

A sample display of the interface configuration for both ends of the serial line follows.

Configuring Serial 1

These commands enable AppleTalk routing in interface serial 1.

Example:

```
interface serial 1
  appletalk address 1544.1
  appletalk zone Twilight
Configuring Serial 2
```

These commands enable AppleTalk on interface serial 2.

Example:

```
interface serial 2
  appletalk address 1544.2
  appletalk zone Twilight
```

You can specify that the interface is enabled using the EXEC command **show appletalk interface** as follows:

```
serial 1 is up, line protocol is up
appleTalk address is 1544.1, Valid
appleTalk zone is Twilight
Serial 2 is up, line protocol is up
AppleTalk address is 1544.2, Valid
AppleTalk zone is Twilight
```

Nonextended (Phase I) AppleTalk Routing over X.25

The configuration of X.25 networks is similar to that for HDLC encapsulation. However, you must completely and explicitly configure *all* network and node numbers in an X.25 environment. Note that all AppleTalk nodes within an X.25 network must be configured with the same AppleTalk network number.

X.25 configuration for AppleTalk involves mapping AppleTalk addresses to X.121 addresses, executed with the X.25 configuration subcommand **x25 map** (see the section “Configuring the Datagram Transport on Commercial X.25 Networks” in the chapter “Configuring Packet-Switched Software”).

Each time a packet is sent to a particular AppleTalk address, that address is looked up in the X.25 map table in order to match it to an X.25 address. The packet is encapsulated in X.25 frames and sent to the X.25 node which is its destination.

The receiving node reassembles the X.25 frames if necessary, then strips the packet of X.25 framing information so that the original AppleTalk datagram can be processed.

In the configuration commands, the keyword **broadcast** (as used at the end of the next example) signals the following to the X.25 software: whenever a broadcast packet is sent, either each or every, but not both map entries with that flag set should receive a copy of the packet. The X.25 protocol does not provide broadcasts; therefore, they must be simulated in this manner when using X.25 as a transport protocol for another protocol that requires broadcasts, such as AppleTalk.

If the X.121 address of the router on the far end of the X.25 network is *123456789012*, and your local X.121 address is *210987654321*, and the two routers are at AppleTalk addresses *7.63* and *7.25*, you would configure these systems in the following way.

Example for First Router:

```
interface serial 0
  appletalk address 7.25
  appletalk zone Twilight
  x25 map appletalk 7.63 123456789012 broadcast
```

Example for Second Router:

```
interface serial 0
  appletalk address 7.63
  appletalk zone Twilight
  x25 map appletalk 7.25 210987654321 broadcast
```

Example for Third Router:

In this example, a third router has the X.121 address *333444555666* and AppleTalk address *7.100*.

```
interface serial 0
  appletalk address 7.100
  appletalk zone Twilight
  x25 map appletalk 7.25 210987654321 broadcast
  x25 map appletalk 7.63 123456789012 broadcast
```

With the addition of the third router, both the original routers need an additional **x25 map** entry:

```
x25 map appletalk 7.100 333444555666 broadcast
```

Extended AppleTalk Routing Network

The following commands illustrate how to configure an extended AppleTalk network.

Example:

```
appletalk routing
!
interface ethernet 0
  appletalk cable-range 69-69 69.128
  appletalk zone Empty Guf
  appletalk zone Underworld
```

This configuration defines the zones *Empty Guf* and *Underworld* from which the router and the nodes may choose to reside. The equal cable range numbers allow compatibility with nonextended AppleTalk networks.

Monitoring the AppleTalk Network

Use the EXEC **show** commands described in this section to obtain displays of activity on the AppleTalk network.

Displaying the Fast-Switching Cache

Use the **show apple cache** command with the extended AppleTalk networks to display the current fast-switching cache. Enter this command at the EXEC prompt:

show apple cache

This display includes the current cache version number and all entries (valid or not). Valid entries are identified by an asterisk (*) in the first column.

Conditions that invalidate the fast-switching cache are as follows:

- Route deleted but not marked bad (and has been used)
- A route which has gone bad (and has been used)
- When you replace a route with a new metric (and it was used)
- When a neighbor transitions from suspect to bad.
- When a node address in the ARP cache changes hardware address.
- When a hardware address changes node address
- When the ARP cache gets flushed
- When an ARP entry is deleted
- When the following configuration commands are entered:
 - After a **no appletalk routing** command
 - After an **appletalk route-cache** command
 - After an AppleTalk **access-list** command
- When the encapsulation for the line changes
- When a port leaves or enters Operational state

Following is a sample display of the **show apple cache** command:

```
AppleTalk routing cache version is 45
  Destination Interface MAC Header
*      4.000 Ethernet1 AA0004007BCC00000C000E8C809B81BE02
*    1544.000 Ethernet1 AA000400013400000C000E8C809B84BE02
*      33.000 Ethernet1 AA000400013400000C000E8C809B84BE02
```

Displaying AppleTalk Interface Information

The **show apple interface** command displays AppleTalk-specific interface information. Enter this command at the EXEC prompt.

show apple interface [*interface*]

The argument *interface* specifies an interface name and number to display a specific interface.

This information displayed by this command includes the extended AppleTalk cable ranges and the current interface mode, (the network verification/discovery mode, for example).

Sample displays of the **show apple interface** command follow.

Nonextended AppleTalk—Normal operation:

```
Ethernet 1 is up, line protocol is up
  AppleTalk address is 666.128, Valid
  AppleTalk zone is Underworld
```

Extended AppleTalk—Normal operation:

```
Ethernet 0 is up, line protocol is up
  AppleTalk cable range is 69-69
  AppleTalk address is 69.128, Valid
  AppleTalk zone is Empty Guf
```

Extended AppleTalk—Verification mode:

```
Ethernet 1 is up, line protocol is up
  AppleTalk routing disabled, Verifying port configuration
  AppleTalk cable range is 666-666
  AppleTalk address is 666.128, Valid
  AppleTalk zone is Underworld
```

Extended AppleTalk—Configuration error:

```
Ethernet 0 is up, line protocol is up
  AppleTalk routing disabled, Port configuration error
  AppleTalk cable range is 70-70
  AppleTalk address is 70.128, Bad
  AppleTalk zone is Empty Guf
```

When you enter the EXEC command **show apple interface** with the *interface* argument, the display looks this:

```
Ethernet 0 is up, line protocol is up
  AppleTalk cable range is 69-69
  AppleTalk address is 69.105, Valid
  AppleTalk zone is "Empty Guf"
  AppleTalk port configuration verified by 69.163
  AppleTalk discarded 3149 packets due to input errors
  AppleTalk discarded 71 packets due to output errors
  AppleTalk route cache is enabled
```

If AppleTalk routing is disabled on an interface, the display looks like this:

```
Ethernet 1 is up, line protocol is up
  AppleTalk protocol processing disabled
```

Displaying Directly Connected Routes

The **show apple neighbor** EXEC command shows the routers that are directly connected, or that are one hop away in the extended AppleTalk network. The command has this syntax:

```
show apple neighbor [neighbor-address]
```

The optional argument *neighbor-address* specifies address of the neighbor of the extended AppleTalk network.

For the command:

```
show apple neighbor
```

The display looks like this:

```
AppleTalk neighbors:

31.86, Ethernet8, uptime 133:28:06, last update 1 sec ago
81.82, Fddi0, uptime 266:11:44, last update 7 secs ago
81.81, Fddi0, uptime 267:30:28, last update 958334 secs ago
  Neighbor is down.
29.200, Ethernet3, uptime 263:45:50, last update 948440 secs ago
  Neighbor has restarted 2 times in 267:59:53.
  Neighbor is down.
81.80, Fddi0, uptime 268:00:08, last update 963617 secs ago
  Neighbor is down.
17.128, Ethernet2, uptime 133:26:43, last update 2 secs ago
  Neighbor has restarted 1 time in 268:00:21.
69.163, Ethernet0, uptime 268:00:25, last update 1 sec ago
```

For the command:

```
show apple neighbor 69.163
```

The display looks like this:

```
Neighbor 69.163, Ethernet0, uptime 268:00:52, last update 7 secs ago
We have sent queries for 299 nets via 214 packets.
Last query was sent 4061 secs ago.

We received 152 replies and 0 extended replies.
We have received queries for 14304 nets via 4835 packets.
We sent 157 replies and 28 extended replies.
We received 0 ZIP notifies.
We received 0 obsolete ZIP commands.
We received 4 miscellaneous ZIP commands.
We received 0 unrecognized ZIP commands.
We have received 92943 routing updates.
Of the 92943 valid updates, 1320 entries were invalid.
We received 1 routing update which were very late.
Last update had 0 extended and 2 nonextended routes.
Last update detail: 2 old
```

Displaying the Network Routing Table

To show the routing table for networks, use the **show apple route** EXEC commands:

```
show apple route [network]
```

show apple route [*interface-name*]

This command displays either the full routing table or just the entry for the optionally specified *network* for both extended and nonextended AppleTalk networks. For the extended AppleTalk networks, the command also displays cable ranges information.

The optional *interface-name* argument specifies an interface name to report on. Displays for both nonextended and extended AppleTalk networks follow.

A sample display for a nonextended AppleTalk network:

```
Codes: R - RTMP derived, C - connected, S - static, 3 routes
C Net 258 directly connected, 1431 uses, Ethernet0, zone Twilight
R Net 6 [1/G] via 258.179, 8 sec, 0 uses, Ethernet0, zone The O
C Net 11 directly connected, 472 uses, Ethernet1, zone No Parking
R Net 2154 [1/G] via 258.179, 8 sec, 6892 uses, Ethernet0, zone LocalTalk
S Net 1111 via 258.144, 0 uses, Ethernet0, no zone set
```

In the above display, the G rating after `Net 6` indicates *good*. Alternate ratings are S for *suspect* and B for *bad*. These ratings are attained from the routing updates which occur at ten-second intervals. A separate and nonsynchronized event occurs at 20-second intervals, checking and flushing the ratings for particular routes that have not been updated. For each 20-second period that passes with no new routing information, a rating will slip from G to S to B; after one minute with no updates, that route will be flushed. Every time the router receives a useful update, the status of the route in question is reset to G. Useful updates are those advertising a route that is as good or better than the one currently in the table.

Following is a sample display for the extended AppleTalk network. Note the cable range display for *Magnolia Estates*:

```
Codes: R - RTMP derived, C - connected, 29 routes in internet

R Net 3 [1/G] via 254.163, 8 sec, Ethernet1, zone Localtalk
C Net 4 directly connected, Ethernet0, zone Twilight
C Net 6 directly connected, Ethernet3, zone Heavenly
R Net 11 [3/G] via 254.163, 8 sec, Ethernet1, zone UDP
R Net 17 [1/G] via 254.163, 8 sec, Ethernet1, zone UDP
R Net 33 [1/G] via 4.129, 1 sec, Ethernet0, zone Twilight
R Net 36 [1/G] via 254.174, 7 sec, Ethernet1, zone idontcare
R Net 55 [1/G] via 254.130, 9 sec, Ethernet1, zone Hospital
R Net 69 [1/G] via 4.129, 1 sec, Ethernet0, zone Empty Guf
R Net 70 [1/G] via 254.247, 2 sec, Ethernet1, zone Empty Guf
C Net 80 directly connected, Ethernet4, zone Light
R Net 99 [2/G] via 4.129, 1 sec, Ethernet0, zone BammBamm
C Net 254 directly connected, Ethernet1, zone Twilight
R Net 890 [2/G] via 4.129, 1 sec, Ethernet0, zone release lab
R Net 901 [2/G] via 4.129, 1 sec, Ethernet0, zone Dave's House
C Net 999-999 directly connected, Serial3, zone Magnolia Estates
R Net 2003 [4/G] via 80.129, 6 sec, Ethernet4, zone Bldg-13
R Net 2004 [2/G] via 80.129, 6 sec, Ethernet4, zone Bldg-17
R Net 2012 [2/G] via 4.130, 7 sec, Ethernet0, zone Bldg-13
R Net 2013 [3/G] via 254.163, 8 sec, Ethernet1, zone UDP
R Net 2024 [4/G] via 80.129, 3 sec, Ethernet4, zone Bldg-17
R Net 3004 [1/G] via 80.129, 3 sec, Ethernet4, zone Bldg-17
R Net 3012 [1/G] via 4.130, 5 sec, Ethernet0, zone Bldg-13
R Net 3024 [4/G] via 80.129, 3 sec, Ethernet4, zone Bldg-17
R Net 3880 [1/G] via 999.2, 0 sec, Serial3, zone Magnolia Estates
```

```
R Net 5002 [2/G] via 80.129, 3 sec, Ethernet4, zone Bldg-17
R Net 5003 [2/G] via 4.130, 5 sec, Ethernet0, zone Bldg-13
R Net 5006 [4/G] via 80.129, 3 sec, Ethernet4, zone Bldg-17
R Net 51489 [3/G] via 4.129, 8 sec, Ethernet0, zone Dave's House
```

[hops/state] state can be one of G:Good, S:Suspect, B:Bad

The next sample shows the result of the **show apple route** command with a specific network.

For the command:

```
show apple route 3880
```

The display looks like this:

```
Codes: R - RTMP derived, C - connected, 29 routes in internet

R Net 3880 [1/G] via 999.2, 7 sec, Serial3, zone Magnolia Estates
Route installed 1:35:37
Current gateway: 999.2, 1 hop away, updated 7 secs ago
Zone list provided by 254.129
Route has been updated since last RTMP was sent
Valid zones: Magnolia Estates
```

For the command:

```
show appletalk route serial 3
```

The display looks like this:

```
Codes: R - RTMP derived, C - connected, 29 routes in internet

C Net 999 directly connected, Serial3, zone Magnolia Estates
R Net 3880 [1/G] via 999.2, 3 sec, Serial3, zone Magnolia Estates
```

Displaying AppleTalk Traffic Information

The EXEC command **show apple traffic** displays AppleTalk-specific traffic information. The command has this syntax:

show apple traffic

The statistics it displays include the total number of packets received, categorized errors, summaries of packets received for the various AppleTalk services (for example, NBP, ZIP, DDP) and for other protocols such as Echo and ARP. Several counters have also been added to monitor extended AppleTalk activity.

Following is a sample display of extended AppleTalk activity.

```

AppleTalk statistics:
  Rcvd: 719 total, 0 checksum errors, 0 bad hop count
        0 local destination, 0 access denied
        2 port disabled
  Bcast: 640 received, 164 sent
  Sent: 164 generated, 0 forwarded
        1736 encapsulation failed, 0 no route
  DDP: 719 long, 0 short, 0 wrong size
  NBP: 5 received, 0 sent, 0 forwarded, 0 lookups
  RTMP: 709 received, 240 sent, 0 requests
  ATP: 0 received
  AMP: 0 received, 0 sent
  ZIP: 3 received, 1659 sent
  Echo: 0 received, 0 illegal
  ARP: 1476 requests, 0 replies, 9 probes
  Lost: 0 no buffers
  Unknown: 0 packets
  Discarded: 1577 wrong encapsulation, 0 bad SNAP discriminator

```

Table 1-1 Show Apple Traffic Field Descriptions

Field	Description
bad hop count	Packet dropped, too many hops.
access denied	Packet dropped, access list didn't permit it.
port disabled	Packet dropped, routing disabled for port (extended AppleTalk only). Occurs because of a configuration error or a packet received while in verification/discovery mode.
encapsulation failed	Packet received for a connected network, but node not found.
wrong size	Physical packet and claimed length disagree.
no buffers	Attempted packet buffer allocation failed.
unknown	Unknown AppleTalk packet type.
wrong encapsulation	Nonextended AppleTalk packet on extended AppleTalk port, or the other way around.
bad SNAP discriminator	Extended AppleTalk packet without Apple discriminator (extended AppleTalk only). Occurs when another AppleTalk device has implemented an obsolete or incorrect packet format.

Displaying Zone Information

The **show apple zone** command displays the zone information table and has this syntax:

```
show apple zone [zonename]
```

Use this command to display which networks comprise each zone for both nonextended and extended AppleTalk networks.

The argument *zonename* specifies the name of the zone you are trying display information on.

In the following sample display, notice the report of cable ranges for the extended zone

“Empty Guf”:

Name	Network(s)
UDP	17 11
Heavenly	1161 6
Hospital	55
Bldg-17	82 81 14 13
CSL EtherTalk	22
Twilight	1544 254 36 33 4
EtherTalk	2
Underworld	666
Magnolia Estates	3880 999
Light	80
LocalTalk	3
Empty Guf	69-69
Total of 12 zones	

Displaying Information About the Sockets

The command **show apple socket** displays information about the process-level processing in all the sockets in the AppleTalk interface. Enter this command at the EXEC prompt:

```
show apple socket [socket-number]
```

When used with the optional *socket-number* argument, it shows information about a specific socket.

The following is the output seen when no socket number is specified:

Socket	Name	Owner	Waiting*/Processed
1	RTMP	AT RTMP	0 99571
2	NIS	AT NBP	0 5425
4	AEP	AT Maintenance	0 1
6	ZIP	AT ZIP	0 2704
253	PingServ	AT Maintenance	0 1
254	Responde	AT Maintenance	0 1

An asterisk (*) indicates the number of packets waiting to be processed.

When a socket is specified, only statistics for that socket are displayed, as seen in following sample output:

6	ZIP	AT ZIP	0 2704
---	-----	--------	--------

Maintaining the AppleTalk Network

Maintaining the AppleTalk network is a simple task. Cisco provides two EXEC commands to clear the different AppleTalk data structures.

Clearing the Neighbor Data Structures

The **clear apple neighbors** command clears the AppleTalk neighbors data structures. Enter this command at the EXEC prompt:

```
clear apple neighbors
```

Clearing the Route Data Structures

The **clear apple routes** command clears the AppleTalk route data structures. Enter this command at the EXEC prompt:

```
clear apple routes
```

The AppleTalk Ping Command

The EXEC **ping** command sends Echo Protocol datagrams to other AppleTalk nodes to verify connectivity and measure round-trip times.

When the **ping** command prompts for a protocol, specify **appletalk**. Default options are indicated with carriage returns. What follows is a sample of using **ping** with the AppleTalk protocol. To abort a ping session, type the escape sequence (by default, Ctrl-^, X).

Sample Session:

```
Protocol [ip]: appletalk
Target Appletalk address: 1024.128
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Verbose [n]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte AppleTalk Echos to 1024.128, timeout is 2 seconds:
!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/4/8 ms
```

The **ping** command uses the characters in Table 1-2 to indicate the success or failure of each packet in the **ping** sequence.

Table 1-2 AppleTalk Ping Characters

Char	Meaning
!	The packet was echoed successfully from the target address.
.	The timeout period expired before an echo was received from the target address.
B	Bad, or malformed echo was received from the target address.
C	An echo was received with a bad DDP checksum.
E	Transmission of the echo packet to the target address failed.
R	The transmission of the echo packet to the target address failed for lack of a route to the target address.

Debugging the AppleTalk Network

The EXEC **debug** commands described in this section are used to troubleshoot the AppleTalk network transactions. Generally, you enter these commands during troubleshooting sessions with Cisco customer engineers.

For each **debug** command, there is a corresponding **undebug** command that turns the display off. Remember that some of these commands can be entered in groups that then display additional information.

debug appletalk

The **debug appletalk** command debugs all start-up messages and protocol routines dedicated to support start-up. This command also debugs global messages such as those regarding neighbors, ports/interfaces, and configuration. The command looks at problems with parts of Appletalk which do not have their own options in other debug commands.

debug apple-arp

The **debug apple-arp** command enables debugging of AppleTalk address resolution protocol. A side-effect of enabling this option is that gleaning MAC information from datagrams is disabled.

debug apple-errors

The **debug apple-errors** command reports information about errors that occur. The information displayed by this command is enhanced by enabling debugging for the specific class of errors that you are interested in. This is similar to **debug apple-packets**.

debug apple-event

The **debug apple-event** command displays debugging information about AppleTalk special events, neighbors becoming reachable/unreachable, and interfaces going up/down. Only significant events (for example, neighbor and/or route changes) are logged. This command is maintained in nonvolatile memory, if present.

apple event-logging

The **apple event-logging** configuration command causes logging of a subset of messages produced by **debug appletalk** command.

debug apple-nbp

The **debug apple-nbp** command enables debugging output from the Name Binding Protocol (NBP) routines.

debug apple-packet

The **debug apple-packet** command enables per-packet debugging output. It reports information online when a packet is received or a transmit is attempted. The command allows watching the types of packets being slow switched. It is roughly equivalent to turning on all the other AppleTalk debugging information. There will be at least one line of debugging output per AppleTalk packet processed.

The **debug apple-packet** command, when invoked in conjunction with the commands **debug apple-routing**, **debug apple-zip**, and **debug apple-nbp**, adds protocol processing information in addition to generic packet details. It reports protocol processing, and successful completion or failure information.

The **debug apple-packet** command, when invoked in conjunction with the command **debug apple-errors**, reports packet level problems such as encapsulation problems. This is the case because **debug apple-errors** is a subset of **debug apple-packets**.

debug apple-routing

The **debug apple-routing** command enables debugging output from the Routing Table Maintenance Protocol (RTMP) routines. This command can be used to monitor acquisition of routes, aging of routing table entries, and advertisement of known routes. It also reports conflicting network numbers on the same network if the network is misconfigured.

debug apple-zip

The **debug apple-zip** command enables debugging output from the Zone Information Protocol routines. This command reports significant events such as discovery of new zones and zone list queries.

AppleTalk Global Configuration Command Summary

This section lists all the global commands used with the AppleTalk interface.

[no] access-list *list* {**permit** | **deny**} *network*

Establishes permit or deny conditions for packets on a specific network interface. The argument *list* is an integer between 600 and 699 and the argument *network* is an AppleTalk network. A *network* argument of -1 represents any network. The **no** form of the command removes an entry from the list.

no appletalk arp

Resets the **arp interval** and **arp retransmit** commands to their default values.

appletalk arp interval *milliseconds*

Specifies the time interval between retransmission of ARP packets. The argument *milliseconds* specifies the interval. The default and minimum value is 33 milliseconds.

appletalk arp retransmit-count *count*

Specifies the number of retransmissions that will be done before abandoning address negotiations and using the selected address. The argument *count* specifies the retransmission count. The minimum value that can be specified is 1 (one); the default is 20.

[no] appletalk checksum

Enables and disables the generation and verification of checksums for all AppleTalk packets. An incoming packet with a nonzero checksum will be verified against that checksum and discarded if in error. By default, checksum verification is enabled.

[no] appletalk event-logging

Causes logging of a subset of messages produced by **debug appletalk** command. The **no** form of the command turns this function off.

[no] apple proxy-npb *network-number zonenumber*

Required for each zone that has a nonextended-only AppleTalk router connected to a network in the zone. The argument *network-number* must be a unique network number which will be advertised via this router as if it were a real network. The argument *zonenumber* is the name of the zone requiring compatibility support. Only one proxy is needed to support a zone, but additional proxies can be defined with different network numbers, if redundancy is desired

[no] appletalk routing

Enables or disables the AppleTalk protocol processing.

[no] appletalk send-rtmp

Allows a router to be placed on a net with AppleTalk, enabled but without being seen. This allows disabling of routine update. The default is to allow the disabling.

[no] appletalk strict-rtmp

Enforces maximum checking of routing packets to insure their validity. The default of this command is to provides maximum checking. The **no** variation disables the maximum checking mode.

[no] appletalk timers *update-interval valid-interval invalid-interval*

Changes the time intervals (in seconds) used in AppleTalk routing. The argument *update-interval* is the time between routing updates sent to other routers on the network; the default is 10 seconds. The argument *valid-interval* is amount of time that the router will consider a route valid without having heard a routing update for that route; the default is 20 seconds, and the value is normally twice the update interval. The argument *invalid-interval* is the amount of time that the router will wait before marking a route invalid; the default is three times the update-interval, or 30 seconds.

AppleTalk Interface Subcommand Summary

This section lists, in alphabetical order, all the interface subcommands used with AppleTalk networks.

[no] appletalk access-group *list*

Assigns an interface to an access list. The argument *list* specifies the appropriate AppleTalk access list. Use the **no** form of the command to remove the list from the interface.

[no] appletalk address *address*

Assigns AppleTalk addresses on the interfaces that will be used for the AppleTalk protocol. This step must be done prior to assigning zone names. Use this subcommand to configure nonextended interfaces.

[no] appletalk cable-range *start-end [network.node]*

Designates an interface as being on an extended AppleTalk network. This range is specified using the *start-end* parameter, which is a pair of decimal numbers between 1 and 65,279, inclusive. The starting and ending addresses can be assigned equal numbers. The optional *network.node* argument specifies the suggested network and node number that will be used first when selecting the AppleTalk address for this interface.

[no] appletalk discovery

Resets the discovery mode and allows a new cable range to be discovered. Use the **no** variation to return the software to the default (off) state.

[no] apple distribute-list *access-list-number in*

Filters input from networks. The argument *access-list-number* is the number of a pre-defined access list. The keyword **in** is used to filter networks received in update.

[no] apple distribute-list *access-list-number out*

Filters output from networks. The argument *access-list-number* is the number of a pre-defined access list. The keyword **out** is used to suppress networks from being sent in updates.

appletalk iptalk-baseport *port-number*

Specifies the UDP port number, which is the beginning of the range of UDP ports used in mapping AppleTalk well-known DDP socket numbers to UDP ports. The argument *port-number* is the UDP port number.

appletalk iptalk *net.node zone*

Encapsulates AppleTalk in IP packets in a manner compatible with the Columbia AppleTalk Package (CAP) IPtalk and the Kinetics IPtalk (KIP) implementations. This command enables IPtalk encapsulation on an interface which already has an configured IP address. The argument *net.node* is a network node number; the argument *zone* the AppleTalk zone.

[no] appletalk zone *zonename*

Sets the zone name for the connected AppleTalk network. This command also specifies the zone name associated with the AppleTalk network for the specified interface. The argument *zonename* specifies the name of the zone for the connected AppleTalk network. The argument is ignored for nonextended AppleTalk. The command is ignored if the specified zone name is not in the zone list. The **no** form of the command deletes a zone name from a zone list or the entire zone list if none is specified.

