Configuring Apollo Domain

The Apollo Domain routing protocol is the native-mode networking protocol for Apollo workstations. This chapter describes how to configure Apollo Domain routing and provides configuration examples. For a complete description of the commands discussed in this chapter, refer to the "*Apollo Domain Commands*" chapter of the *Router Products Command Reference* publication.

Cisco's Implementation of Apollo Domain

The Cisco routing software implementation supports packet forwarding and routing for the Apollo Domain network protocols on Ethernet, Fiber Distributed Data Interface (FDDI), and serial interfaces using High-level Data Link Control (HDLC) or X.25 encapsulation. The software implementation does not support direct attachment to the 12-MB Domain Token Ring.

The following restrictions apply to the Cisco implementation of Apollo Domain:

- When both bridging and Apollo Domain routing are enabled on an Ethernet network, you must specify an Ethernet type code access list that filters out datagrams with the Apollo Domain type code (hexadecimal 8019). This restriction applies to MCI cards running microcode Version 1.5 or earlier.

- You must set IP addresses on all networks that use the Internet Protocol (IP) address resolution protocol (ARP) (for example, Ethernet and FDDI). This is necessary because Domain ARP (sometimes called D-ARP) uses the same Ethernet type value as IP ARP.

- The Cisco implementation of the Apollo Domain routing assumes that ARP can be used to locate workstations on the local cable. The following workstations and versions of the Apollo operating system support Domain ARP:

  — DN3000 and DN3010 nodes need Version 9.7.4.1, which is available from local Apollo field offices on patch tape 186.

  — DN3500, 4000, and 4500 nodes need Version 9.7.5.1, which is available from local Apollo field offices on patch tape 185.

  — Version 9.7, which provides ARP for DN5*xx*-T nodes, needs Version 9.7.4.b101. No patch is available for these workstations. The software is provided only on a DECnet tape.

Note   Version 10.0 of the Apollo Domain operating system does not provide ARP. You must migrate to Version 10.1 and later versions before you can operate with Cisco routers. Cisco routers support neither the **rtchk** and **lcnode** commands nor D-ARP in Apollo's 802.5 implementation.

Apollo Domain Addresses

Apollo Domain network addresses are 32-bit quantities represented in hexadecimal numbers in the format *network.host*. Each host has a single address that is used for all its network connections.

The network number is a 12-bit number, expressed in hexadecimal, that identifies a physical network. Network numbers must be unique throughout an Apollo Domain internetwork.

The host number is a 20-bit quantity expressed in hexadecimal. An Apollo Domain host can have interfaces on more than one physical network (Ethernet, Domain Token Ring, serial line, and so on).

In the following example of a network address, the number 5fe identifies a physical network and the number 1293c identifies a host (see Figure 1-1):

```
5fe.1293c
```

Figure 1-1   Apollo Domain Addresses
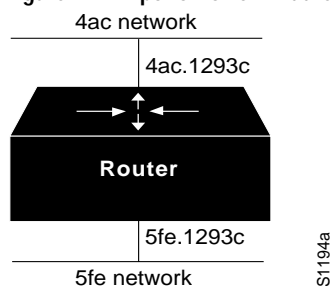
Apollo Domain Configuration Task List

To configure Apollo Domain routing, complete the following tasks. At a minimum, you must enable routing.

- Enable Apollo Domain routing (page 11-3)
- Control access to Apollo Domain networks (page 11-3)
- Tune Apollo Domain network performance (page 11-4)
- Configure Apollo Domain over WANs (page 11-5)
- Monitor the Apollo Domain network (page 11-5)

This chapter describes how to perform these tasks. Configuration examples are provided at the end of the chapter.

Enable Apollo Domain Routing

To enable the Apollo Domain routing protocol, you must enable it both on the router and on each interface. To do this, perform the following tasks:

Task		Command
Step 1	Enter global configuration mode.	See Table 2-1.
Step 2	Enable Apollo Domain routing on the router.	**apollo routing** *host*
Step 3	Enter interface configuration mode.	See Table 2-1.
Step 4	Enable Apollo Domain routing on an interface.	**apollo network** *number*

For an example of how to enable Apollo Domain routing, see the section "Configuring Apollo Domain Routing Example" later in this chapter.

Control Access to the Apollo Domain Network

To control access to Apollo Domain networks, you create access lists and then apply them to individual interfaces. Apollo Domain access lists control access based on a range of network numbers. The conditions defined in access lists are applied to outgoing routed packets.

Keep the following in mind when configuring Apollo Domain network access control:

- Access lists entries are evaluated in the order you enter them, and the first matching entry is used. To improve performance, place the most commonly matched entries near the beginning of the access list.

- An implicit *deny everything* entry is defined at the end of an access list unless you include an explicit *permit everything* entry at the end of the list.

- All new entries to an existing list are placed at the end of the list. You cannot add an entry to the middle of a list. This means that if you have previously included an explicit *permit everything* entry, new entries will never be scanned. The solution is to delete the access list and retype it with the new entries.

To create an access list, perform the following task in global configuration mode:

Task	Command
Create an Apollo Domain access list.	**apollo access-list** *access-list-name* {**deny**	**permit**} [*firstnet-*]*lastnet.host* [*wildcard-mask*]

To apply an access list to an interface and activate it on that interface, perform the following task in interface configuration mode:

Task	Command
Apply an access list name to an interface.	**apollo access-group** *access-list-name*

For an example of creating and applying an access list, see the section "Access List Example" later in this chapter.

Use the **show apollo interface** command to display the filters defined on an interface.

Tune Apollo Domain Network Performance

To tune Apollo Domain network performance, perform one or more of the following tasks:

- Configure static routes
- Set routing table update timers
- Set maximum paths

Configure Static Routes

The router uses metrics to determine the best path over which packets should be transmitted. However, you may want to add static routes to the routing table to explicitly specify paths to certain destinations. Static routes always override any paths determined by metrics.

To add a static route to the router's routing table, perform the following task in global configuration mode:

Task	Command
Add a static route to the routing table.	**apollo route** *destination-network network.address*

Set Routing Table Update Timers

You can set how often the entries in the routing table are updated. Note, however, that you should set this interval only in a configuration in which all routers are our routers.

To set how often the entries in the routing table are updated, perform the following task in interface configuration mode:

Task	Command
Set the interval after which the routing table should be updated.	**apollo update-time** *interval*

For an example of setting routing table update timers, see the section "Routing Table Update Timer Example" later in this chapter.

Set the Maximum Paths

You can set the maximum number of equal-cost, parallel paths to a destination. (Note that when paths have differing costs, the router chooses lower-cost routes in preference to higher-cost routes.) The router distributes output on a packet-by-packet basis in round-robin fashion. That is, the first packet is sent along the first path, the second packet along the second path, and so on. If the final path is reached before all packets are sent, the next packet is sent to the first path, the next to the second path, and so on. This round-robin scheme is used regardless of whether fast switching is enabled.

Limiting the number of equal-cost paths can save memory on routers with limited memory or very large configurations. Additionally, in networks with a large number of multiple paths and systems with limited ability to cache out-of-sequence packets, performance might suffer when traffic is split between many paths.

To set the maximum number of paths on the router, perform the following task in global configuration mode:

Task	Command
Set the maximum number of equal-cost paths to a destination.	**apollo maximum-paths** *paths*

Configure Apollo Domain over WANs

You can configure Apollo Domain over X.25 networks. To do this, configure the appropriate address mappings as described in the chapter,"*Configuring X.25 and LAPB*".

Monitor the Apollo Domain Network

To monitor an Apollo Domain network, perform one or more of the following tasks at the EXEC prompt:

Task	Command
List the entries in the Apollo Domain ARP table.	**show apollo arp**
Display the status of the Apollo Domain interfaces configured in the router and the parameters configured on each interface.	**show apollo interface** [*interface unit*]
List the entries in the Apollo Domain routing table.	**show apollo route** [*network*]
Display information about the number and type of Apollo Domain packets transmitted and received.	**show apollo traffic**

Apollo Domain Configuration Examples

Use the following configuration examples to help in configuring Apollo Domain routing on your router:

- Configuring Apollo Domain routing example (page 11-5)

- Access list example (page 11-6)

- Routing table update timer example (page 11-6)

Configuring Apollo Domain Routing Example

The following is an example of configuring Apollo Domain routing on a router with two Ethernet interfaces. The first set of commands enables the Apollo Domain routing protocol, assigns an Apollo Domain network address, and assigns network numbers to two Ethernet interfaces.

The second set of commands does the following:

- Assign the router whose host number is 23d5a to handle all traffic routed over network 35.

- Allow traffic to be routed over a maximum of four paths.

- Set the timer to update the routing table for Ethernet interface 0 every 40 seconds.

- Create an access list called doc that denies access to networks 2a through 2f, and apply this access list to Ethernet interface 1. This means that packets destined for these networks that are sent out Ethernet interface 1 will be blocked.

```
!
apollo routing 23d5a
interface ethernet 0
apollo network 5f
interface ethernet 1
apollo network 4e
!
apollo route 35 23d5a
apollo maximum-paths 4
interface ethernet 0
apollo update-time 40
apollo access-list doc deny 2a-2f
interface ethernet 1
apollo access-group doc
```

Access List Example

The following example creates an Apollo Domain access list and applies it to Ethernet interface 0. In this example, the first line denies access to networks 3a through 3f, the second line denies access to the host 5fe.1293.c, and the third line permits access to all other networks and hosts. The access list conditions will be applied to all routed packets going out Ethernet interface 0.

```
apollo access-list eng deny 3a-3f.0 ffff
apollo access-list eng deny 5fe.1293c
apollo access-list eng permit -1.0 ffff
!
interface ethernet 0
apollo access-group eng
```

Routing Table Update Timer Example

The following example sets the update times on two interfaces in the router. The update timer granularity would be 20 seconds, because this is the lowest value specified.

```
interface serial 0
apollo update-time 40
interface ethernet 0
apollo update-time 20
interface ethernet 1
apollo update-time 25
```