IP Addressing: IPv4 Addressing Configuration Guide, Cisco IOS XE 16.7.x

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

CHAPTER 1

Read Me First

Important Information about Cisco IOS XE 16

Effective Cisco IOS XE Release 3.7.0E (for Catalyst Switching) and Cisco IOS XE Release 3.17S (for Access and Edge Routing) the two releases evolve (merge) into a single version of converged release—the Cisco IOS XE 16—providing one release covering the extensive range of access and edge products in the Switching and Routing portfolio.

Feature Information

Use Cisco Feature Navigator to find information about feature support, platform support, and Cisco software image support. An account on Cisco.com is not required.

Related References

- Cisco IOS Command References, All Releases

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see What's New in Cisco Product Documentation.

To receive new and revised Cisco technical content directly to your desktop, you can subscribe to the What's New in Cisco Product Documentation RSS feed. RSS feeds are a free service.

CHAPTER 2

Configuring IPv4 Addresses

This chapter contains information about, and instructions for configuring IPv4 addresses on interfaces that are part of a networking device.

Note All further references to IPv4 addresses in this document use only IP in the text, not IPv4.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

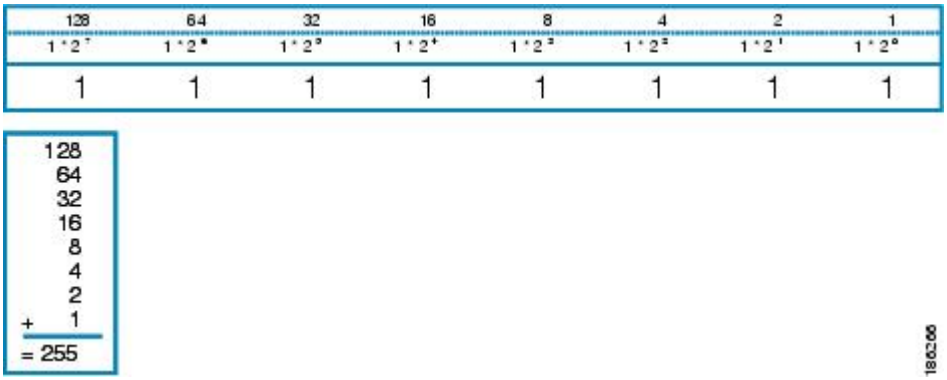
Information About IP Addresses

Binary Numbering

IP addresses are 32 bits long. The 32 bits are divided into four octets (8-bits). A basic understanding of binary numbering is very helpful if you are going to manage IP addresses in a network because changes in the values of the 32 bits indicate either a different IP network address or IP host address.

A value in binary is represented by the number (0 or 1) in each position multiplied by the number 2 to the power of the position of the number in sequence, starting with 0 and increasing to 7, working right to left. The figure below is an example of an 8-digit binary number.

Figure 1: Example of an 8-digit Binary Number



The figure below provides binary to decimal number conversion for 0 through 134.

Figure 2: Binary to Decimal Number Conversion for 0 to 134

00000000 = 000	00011011 = 027	00110110 = 054	01010001 = 081	01101100 = 108
00000001 = 001	00011100 = 028	00110111 = 055	01010010 = 082	01101101 = 109
00000010 = 002	00011101 = 029	00111000 = 056	01010011 = 083	01101110 = 110
00000011 = 003	00011110 = 030	00111001 = 057	01010100 = 084	01101111 = 111
00000100 = 004	00011111 = 031	00111010 = 058	01010101 = 085	01110000 = 112
00000101 = 005	00100000 = 032	00111011 = 059	01010110 = 086	01110001 = 113
00000110 = 006	00100001 = 033	00111100 = 060	01010111 = 087	01110010 = 114
00000111 = 007	00100010 = 034	00111101 = 061	01011000 = 088	01110011 = 115
00001000 = 008	00100011 = 035	00111110 = 062	01011001 = 089	01110100 = 116
00001001 = 009	00100100 = 036	00111111 = 063	01011010 = 090	01110101 = 117
00001010 = 010	00100101 = 037	01000000 = 064	01011011 = 091	01110110 = 118
00001011 = 011	00100110 = 038	01000001 = 065	01011100 = 092	01110111 = 119
00001100 = 012	00100111 = 039	01000010 = 066	01011101 = 093	01111000 = 120
00001101 = 013	00101000 = 040	01000011 = 067	01011110 = 094	01111001 = 121
00001110 = 014	00101001 = 041	01000100 = 068	01011111 = 095	01111010 = 122
00001111 = 015	00101010 = 042	01000101 = 069	01100000 = 096	01111011 = 123
00010000 = 016	00101011 = 043	01000110 = 070	01100001 = 097	01111100 = 124
00010001 = 017	00101100 = 044	01000111 = 071	01100010 = 098	01111101 = 125
00010010 = 018	00101101 = 045	01001000 = 072	01100011 = 099	01111110 = 126
00010011 = 019	00101110 = 046	01001001 = 073	01100100 = 100	01111111 = 127
00010100 = 020	00101111 = 047	01001010 = 074	01100101 = 101	10000000 = 128
00010101 = 021	00110000 = 048	01001011 = 075	01100110 = 102	10000001 = 129
00010110 = 022	00110001 = 049	01001100 = 076	01100111 = 103	10000010 = 130
00010111 = 023	00110010 = 050	01001101 = 077	01101000 = 104	10000011 = 131
00011000 = 024	00110011 = 051	01001110 = 078	01101001 = 105	10000100 = 132
00011001 = 025	00110100 = 052	01001111 = 079	01101010 = 106	10000101 = 133
00011010 = 026	00110101 = 053	01010000 = 080	01101011 = 107	10000110 = 134

186267

The figure below provides binary to decimal number conversion for 135 through 255.

Figure 3: Binary to Decimal Number Conversion for 135 to 255

10000111 = 135	10100010 = 162	10111101 = 189	11011000 = 216	11110011 = 243
10001000 = 136	10100011 = 163	10111110 = 190	11011001 = 217	11110100 = 244
10001001 = 137	10100100 = 164	10111111 = 191	11011010 = 218	11110101 = 245
10001010 = 138	10100101 = 165	11000000 = 192	11011011 = 219	11110110 = 246
10001011 = 139	10100110 = 166	11000001 = 193	11011100 = 220	11110111 = 247
10001100 = 140	10100111 = 167	11000010 = 194	11011101 = 221	11111000 = 248
10001101 = 141	10101000 = 168	11000011 = 195	11011110 = 222	11111001 = 249
10001110 = 142	10101001 = 169	11000100 = 196	11011111 = 223	11111010 = 250
10001111 = 143	10101010 = 170	11000101 = 197	11100000 = 224	11111011 = 251
10010000 = 144	10101011 = 171	11000110 = 198	11100001 = 225	11111100 = 252
10010001 = 145	10101100 = 172	11000111 = 199	11100010 = 226	11111101 = 253
10010010 = 146	10101101 = 173	11001000 = 200	11100011 = 227	11111110 = 254
10010011 = 147	10101110 = 174	11001001 = 201	11100100 = 228	11111111 = 255
10010100 = 148	10101111 = 175	11001010 = 202	11100101 = 229	
10010101 = 149	10110000 = 176	11001011 = 203	11100110 = 230	
10010110 = 150	10110001 = 177	11001100 = 204	11100111 = 231	
10010111 = 151	10110010 = 178	11001101 = 205	11101000 = 232	
10011000 = 152	10110011 = 179	11001110 = 206	11101001 = 233	
10011001 = 153	10110100 = 180	11001111 = 207	11101010 = 234	
10011010 = 154	10110101 = 181	11010000 = 208	11101011 = 235	
10011011 = 155	10110110 = 182	11010001 = 209	11101100 = 236	
10011100 = 156	10110111 = 183	11010010 = 210	11101101 = 237	
10011101 = 157	10111000 = 184	11010011 = 211	11101110 = 238	
10011110 = 158	10111001 = 185	11010100 = 212	11101111 = 239	
10011111 = 159	10111010 = 186	11010101 = 213	11110000 = 240	
10100000 = 160	10111011 = 187	11010110 = 214	11110001 = 241	
10100001 = 161	10111100 = 188	11010111 = 215	11110010 = 242	

186271

IP Address Structure

An IP host address identifies a device to which IP packets can be sent. An IP network address identifies a specific network segment to which one or more hosts can be connected. The following are characteristics of IP addresses:

- IP addresses are 32 bits long
- IP addresses are divided into four sections of one byte (octet) each
- IP addresses are typically written in a format known as dotted decimal

The table below shows some examples of IP addresses.

Table 1: Examples of IP Addresses

IP Addresses in Dotted Decimal	IP Addresses in Binary
10.34.216.75	00001010.00100010.11011000.01001011
172.16.89.34	10101100.00010000.01011001.00100010
192.168.100.4	11000000.10101000.01100100.00000100

Note The IP addresses in the table above are from RFC 1918, *Address Allocation for Private Internets* . These IP addresses are not routable on the Internet. They are intended for use in private networks. For more information on RFC1918, see http://www.ietf.org/rfc/rfc1918.txt .

IP addresses are further subdivided into two sections known as network and host. The division is accomplished by arbitrarily ranges of IP addresses to classes. For more information see RFC 791 Internet Protocol at http://www.ietf.org/rfc/rfc0791.txt .

IP Address Classes

In order to provide some structure to the way IP addresses are assigned, IP addresses are grouped into classes. Each class has a range of IP addresses. The range of IP addresses in each class is determined by the number of bits allocated to the network section of the 32-bit IP address. The number of bits allocated to the network section is represented by a mask written in dotted decimal or with the abbreviation /*n* where *n* = the numbers of bits in the mask.

The table below lists ranges of IP addresses by class and the masks associated with each class. The digits in bold indicate the network section of the IP address for each class. The remaining digits are available for host IP addresses. For example, IP address 10.90.45.1 with a mask of 255.0.0.0 is broken down into a network IP address of 10.0.0.0 and a host IP address of 0.90.45.1.

Table 2: IP Address Ranges by Class with Masks

Class	Range
A (range/mask in dotted decimal)	**0** .0.0.0 to 127.0.0.0/8 (255.0.0.0)
A (range in binary)	**00000000** .00000000.00000000.00000000 to**01111111**.00000000.00000000.00000000
A (mask in binary)	11111111.00000000.00000000.00000000/8
B (range/mask in dotted decimal)	**128** .0.0.0 to **191**.**255**.0.0/16 (255.255.0.0)
B (range in binary)	**10000000** .**00000000**.00000000.00000000 to**10111111**.**11111111**.00000000.00000000
B (mask in binary)	**11111111** .**11111111**.00000000.00000000/16
C (range/mask in dotted decimal)	**192** .**0**.**0**.0 to **223**.**255**.**255**.0/24 (255.255.255.0)
C (range in binary)	**11000000** .**00000000**.**00000000**.00000000 to**11011111**.**11111111**.**11111111**.00000000
C (mask in binary)	11111111.11111111.11111111.0000000/24
D[1] (range/mask in dotted decimal)	**224** .**0**.**0**.**0** to **239**.**255**.**255**.**255**/32 (255.255.255.255)

Class	Range
D (range in binary)	**11100000 .00000000.00000000.00000000** to**11101111.11111111.11111111.11111111**
D (mask in binary)	11111111.11111111.11111111.11111111/32
E[2] (range/mask in dotted decimal)	**240 .0.0.0** to **255.255.255.255**/32 (255.255.255.255)
E (range in binary)	**11110000 .00000000.00000000.00000000** to**11111111.11111111.11111111.11111111**
E (mask in binary)	11111111.11111111.11111111.11111111/32

1 Class D IP addresses are reserved for multicast applications.
2 Class E IP addresses are reserved for broadcast traffic.

Note Some IP addresses in these ranges are reserved for special uses. For more information refer to RFC 3330, *Special-Use IP Addresses* , at http://www.ietf.org/rfc/rfc3330.txt .

When a digit that falls within the network mask changes from 1 to 0 or 0 to 1 the network address is changed. For example, if you change 10101100.00010000.01011001.00100010/16 to 10101100.00110000.01011001.00100010/16 you have changed the network address from 172.16.89.34/16 to 172.48.89.34/16.

When a digit that falls outside the network mask changes from 1 to 0 or 0 to 1 the host address is changed. For example, if you change 10101100.00010000.01011001.00100010/16 to 10101100.00010000.01011001.00100011/16 you have changed the host address from 172.16.89.34/16 to 172.16.89.35/16.

Each class of IP address supports a specific range of IP network addresses and IP host addresses. The range of IP network addresses available for each class is determined with the formula 2 to the power of the number of available bits. In the case of class A addresses, the value of the first bit in the 1st octet (as shown in the table above) is fixed at 0. This leaves 7 bits for creating additional network addresses. Therefore there are 128 IP network addresses available for class A (27 = 128).

The number of IP host addresses available for an IP address class is determined by the formula 2 to the power of the number of available bits minus 2. There are 24 bits available in a class A addresses for IP host addresses. Therefore there are 16,777,214 IP hosts addresses available for class A ((224) - 2 = 16,777,214)).

Note The 2 is subtracted because there are 2 IP addresses that cannot be used for a host. The all 0's host address cannot be used because it is the same as the network address. For example, 10.0.0.0 cannot be both a IP network address and an IP host address. The all 1's address is a broadcast address that is used to reach all hosts on the network. For example, an IP datagram addressed to 10.255.255.255 will be accepted by every host on network 10.0.0.0.

The table below shows the network and host addresses available for each class of IP address.

Table 3: Network and Host Addresses Available for Each Class of IP Address

Class	Network Addresses	Host Addresses
A	128	16,777,214
B	16,384[3]	65534
C	2,097,152[4]	254

[3] Only 14 bits are available for class B IP network addresses because the first 2 bits are fixed at 10 as shown in Table 2 .
[4] Only 21 bits are available for class C IP network addresses because the first 3 bits are fixed at 110 as shown in Table 2 .

IP Network Subnetting

The arbitrary subdivision of network and host bits in IP address classes resulted in an inefficient allocation of IP space. For example, if your network has 16 separate physical segments you will need 16 IP network addresses. If you use 16 class B IP network addresses, you would be able to support 65,534 hosts on each of the physical segments. Your total number of supported host IP addresses is 1,048,544 (16 * 65,534 = 1,048,544). Very few network technologies can scale to having 65,534 hosts on a single network segment. Very few companies need 1,048,544 IP host addresses. This problem required the development of a new strategy that permitted the subdivision of IP network addresses into smaller groupings of IP subnetwork addresses. This strategy is known as subnetting.

If your network has 16 separate physical segments you will need 16 IP subnetwork addresses. This can be accomplished with one class B IP address. For example, start with the class B IP address of 172.16.0.0 you can reserve 4 bits from the third octet as subnet bits. This gives you 16 subnet IP addresses 24 = 16. The table below shows the IP subnets for 172.16.0.0/20.

Table 4: Examples of IP Subnet Addresses using 172.16.0.0/20

Number	IP Subnet Addresses in Dotted Decimal	IP Subnet Addresses in Binary
0[5]	172.16.0.0	10101100.00010000.00000000.00000000
1	172.16.16.0	10101100.00010000.00010000.00000000
2	172.16.32.0	10101100.00010000.00100000.00000000
3	172.16.48.0	10101100.00010000.00110000.00000000
4	172.16.64.0	10101100.00010000.01000000.00000000
5	172.16.80.0	10101100.00010000.01010000.00000000
6	172.16.96.0	10101100.00010000.01100000.00000000
7	172.16.112.0	10101100.00010000.01110000.00000000

Number	IP Subnet Addresses in Dotted Decimal	IP Subnet Addresses in Binary
8	172.16.128.0	10101100.00010000.10000000.00000000
9	172.16.144.0	10101100.00010000.10010000.00000000
10	172.16.160.0	10101100.00010000.10100000.00000000
11	172.16.176.0	10101100.00010000.10110000.00000000
12	172.16.192.0	10101100.00010000.11000000.00000000
13	172.16.208.0	10101100.00010000.11010000.00000000
14	172.16.224.0	10101100.00010000.11100000.00000000
15	172.16.240.0	10101100.00010000.11110000.00000000

5 The first subnet that has all of the subnet bits set to 0 is referred to as subnet 0 . It is indistinguishable from the network address and must be used carefully.

When a digit that falls within the subnetwork (subnet) mask changes from 1 to 0 or 0 to 1 the subnetwork address is changed. For example, if you change 10101100.00010000.01011001.00100010/20 to 10101100.00010000.01111001.00100010/20 you have changed the network address from 172.16.89.34/20 to 172.16.121.34/20.

When a digit that falls outside the subnet mask changes from 1 to 0 or 0 to 1 the host address is changed. For example, if you change 10101100.00010000.01011001.00100010/20 to 10101100.00010000.01011001.00100011/20 you have changed the host address from 172.16.89.34/20 to 172.16.89.35/20.

Timesaver To avoid having to do manual IP network, subnetwork, and host calculations, use one of the free IP subnet calculators available on the Internet.

Some people get confused about the terms network address and subnet or subnetwork addresses and when to use them. In the most general sense the term network address means “the IP address that routers use to route traffic to a specific network segment so that the intended destination IP host on that segment can receive it”. Therefore the term network address can apply to both non-subnetted and subnetted IP network addresses. When you are troubleshooting problems with forwarding traffic from a router to a specific IP network address that is actually a subnetted network address, it can help to be more specific by referring to the destination network address as a subnet network address because some routing protocols handle advertising subnet network routes differently from network routes. For example, the default behavior for RIP v2 is to automatically summarize the subnet network addresses that it is connected to their non-subnetted network addresses (172.16.32.0/24 is advertised by RIP v2 as 172.16.0.0/16) when sending routing updates to other routers. Therefore the other routers might have knowledge of the IP network addresses in the network, but not the subnetted network addresses of the IP network addresses.

Tip The term IP address space is sometimes used to refer to a range of IP addresses. For example, “We have to allocate a new IP network address to our network because we have used all of the available IP addresses in the current IP address space”.

IP Network Address Assignments

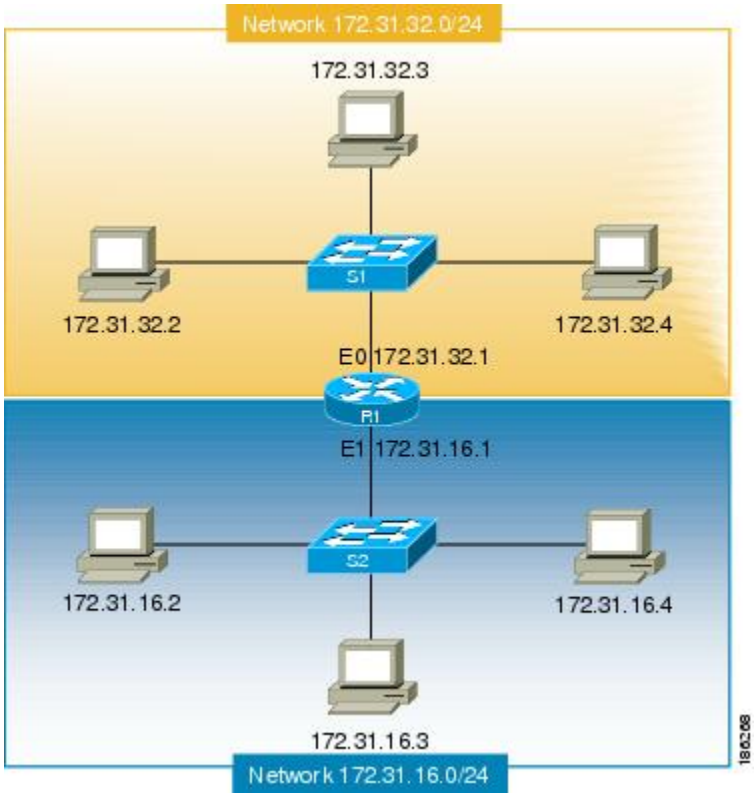
Routers keep track of IP network addresses to understand the network IP topology (layer 3 of the OSI reference model) of the network to ensure that IP traffic can be routed properly. In order for the routers to understand the network layer (IP) topology, every individual physical network segment that is separated from any other physical network segment by a router must have a unique IP network address.

The figure below shows an example of a simple network with correctly configured IP network addresses. The routing table in R1 looks like the table below.

Table 5: Routing Table for a Correctly Configured Network

Interface Ethernet 0	Interface Ethernet 1
172.31.32.0/24 (Connected)	172.31.16.0/24 (Connected)

Figure 4: Correctly Configured Network

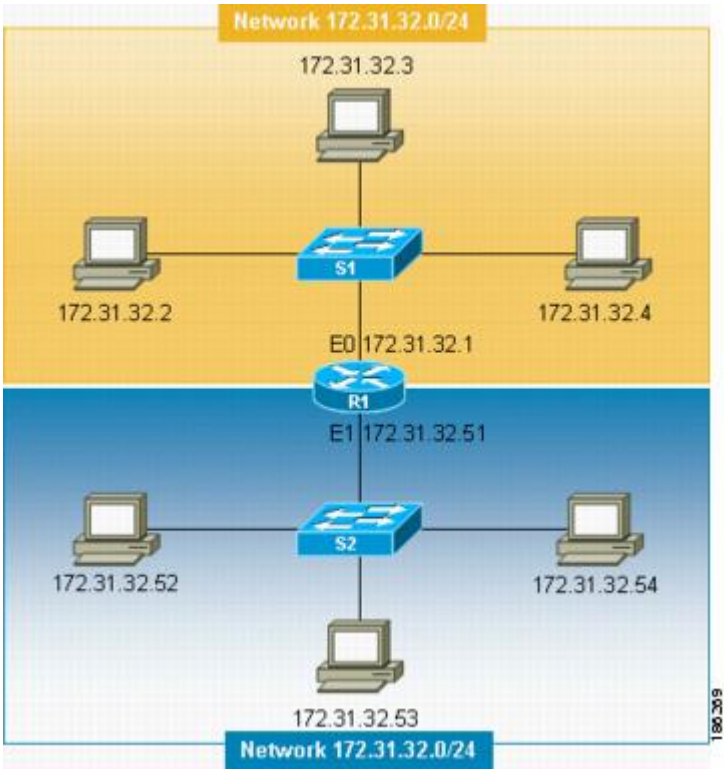


The figure below shows an example of a simple network with incorrectly configured IP network addresses. The routing table in R1 looks like the table below. If the PC with IP address 172.31.32.3 attempts to send IP traffic to the PC with IP address 172.31.32.54, router R1 cannot determine which interface that the PC with IP address 172.31.32.54 is connected to.

Table 6: Routing Table in Router R1 for an Incorrectly Configured Network (Example 1)

Ethernet 0	Ethernet 1
172.31.32.0/24 (Connected)	172.31.32.0/24 (Connected)

Figure 5: Incorrectly Configured Network (Example 1)



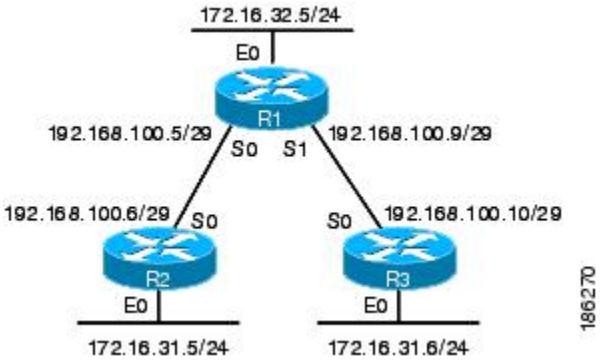
To help prevent mistakes as shown in the figure above, Cisco IOS-based networking devices will not allow you to configure the same IP network address on two or more interfaces in the router when IP routing is enabled.

The only way to prevent the mistake shown in the figure below, where 172.16.31.0/24 is used in R2 and R3, is to have very accurate network documentation that shows where you have assigned IP network addresses.

Table 7: Routing Table in Router R1 for an Incorrectly Configured Network (Example 2)

Ethernet 0	Serial 0	Serial 1
172.16.32.0/24 (Connected)	192.168.100.4/29 (Connected) 172.16.31.0/24 RIP	192.168.100.8/29 (Connected) 172.16.31.0/24 RIP

Figure 6: Incorrectly Configured Network (Example 2)



For a more thorough explanation of IP routing, see the "Related Documents" section for a list of documents related to IP routing.

Classless Inter-Domain Routing

Due to the continuing increase in internet use and the limitations on how IP addresses can be assigned using the class structure shown in the table above, a more flexible method for allocating IP addresses was required. The new method is documented in RFC 1519 *Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*. CIDR allows network administrators to apply arbitrary masks to IP addresses to create an IP addressing plan that meets the requirements of the networks that they administrate.

For more information on CIDR, refer to RFC 1519 at http://www.ietf.org/rfc/rfc1519.txt.

Prefixes

The term prefix is often used to refer to the number of bits of an IP network address that are of importance for building routing tables. If you are using only classful (strict adherence to A, B, and C network address boundaries) IP addresses, the prefixes are the same as the masks for the classes of addresses. For example, using classful IP addressing, a class C IP network address such as 192.168.10.0 uses a 24-bit mask (/24 or 255.255.255.0) and can also be said to have a 24-bit prefix.

If you are using CIDR, the prefixes are arbitrarily assigned to IP network addresses based on how you want to populate the routing tables in your network. For example, a group of class C IP addresses such as 192.168.10.0, 192.168.11.0, 192.168.12.0, 192.168.13.0 can be advertised as a single route to 192.168.0.0 with a 16-bit prefix (192.168.0.0/16). This results in a 4:1 reduction in the number of routes that the routers in your network need to manage.

How to Configure IP Addresses

Establishing IP Connectivity to a Network by Assigning an IP Address to an Interface

Perform this task to configure an IP address on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no shutdown**
5. **ip address** *ip-address mask*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 4	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.

	Command or Action	Purpose
Step 5	**ip address** *ip-address* *mask* **Example:** `Router(config-if)# ip address 172.16.16.1 255.255.240.0`	Configures the IP address on the interface.
Step 6	**end** **Example:** `Router(config-if)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

The following commands can help troubleshoot IP addressing:

- **show ip interface** --Displays the IP parameters for the interface.
- **show ip route connected** --Displays the IP networks the networking device is connected to.

Increasing the Number of IP Hosts that Are Supported on a Network by Using Secondary IP Addresses

If you have a situation in which you need to connect more IP hosts to a network segment and you have used all of the available IP host addresses for the subnet to which you have assigned the segment, you can avoid having to readdress all of the hosts with a different subnet by adding a second IP network address to the network segment.

Perform this task to configure a secondary IP address on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type* *number*
4. **no shutdown**
5. **ip address** *ip-address* *mask*
6. **ip address** *ip-address* *mask* **secondary**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 4	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.
Step 5	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 172.16.16.1 255.255.240.0`	Configures the IP address on the interface.
Step 6	**ip address** *ip-address mask* **secondary** **Example:** `Router(config-if)# ip address 172.16.32.1 255.255.240.0 secondary`	Configures the secondary IP address on the interface.
Step 7	**end** **Example:** `Router(config-if)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

The following commands can help troubleshoot IP addressing:

- **show ip interface** --Displays the IP parameters for the interface.
- **show ip route connected** --Displays the IP networks the networking device is connected to.

What to Do Next

If your network has two or more routers and you have already configured a routing protocol, make certain that the other routers can reach the new IP network that you assigned. You might need to modify the configuration for the routing protocol on the router so that it advertises the new network. Consult the *Cisco IOS IP Routing: Protocol-Independent Configuration Guide* for information on configuring routing protocols.

Maximizing the Number of Available IP Subnets by Allowing the Use of IP Subnet Zero

If you using subnetting in your network and you are running out of network addresses, you can configure your networking device to allow the configuration of subnet zero. This adds one more usable network address for every subnet in your IP addressing scheme. The table above shows the IP subnets (including subnet 0) for 172.16.0.0/20.

Perform this task to enable the use of IP subnet zero on your networking device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip subnet-zero**
4. **interface** *type number*
5. **no shutdown**
6. **ip address** *ip-address mask*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ip subnet-zero** **Example:** `Router(config)# ip subnet-zero`	Enables the use of IP subnet zero.
Step 4	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 5	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.
Step 6	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 172.16.0.1 255.255.240.0`	Configures the subnet zero IP address on the interface.
Step 7	**end** **Example:** `Router(config-if)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

The following commands can help troubleshoot IP addressing:

- **show ip interface** --Displays the IP parameters for the interface.
- **show ip route connected** --Displays the IP networks the networking device is connected to.

Specifying the Format of Network Masks

By default, **show** commands display an IP address and then its netmask in dotted decimal notation. For example, a subnet would be displayed as 131.108.11.55 255.255.255.0.

You might find it more convenient to display the network mask in hexadecimal format or bit count format instead. The hexadecimal format is commonly used on UNIX systems. The previous example would be displayed as 131.108.11.55 0XFFFFFF00.

The bit count format for displaying network masks is to append a slash (/) and the total number of bits in the netmask to the address itself. The previous example would be displayed as 131.108.11.55/24.

Specifying the Format in Which Netmasks Appear for the Current Session

Perform this task to specify the format in which netmasks appear for the current session.

SUMMARY STEPS

1. **enable**
2. **term ip netmask-format** {**bitcount** | **decimal** | **hexadecimal**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**term ip netmask-format** {**bitcount** | **decimal** | **hexadecimal**} **Example:** `Router# term ip netmask-format hexadecimal`	Specifies the format the router uses to display network masks.

Specifying the Format in Which Netmasks Appear for an Individual Line

Perform this task to specify the format in which netmasks appear for an individual line.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty** *first last*
4. **term ip netmask-format** {**bitcount** | **decimal** | **hexadecimal**}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**line vty** *first last* **Example:** `Router(config)# line vty 0 4`	Enters line configuration mode for the range of lines specified by the *first* and *last* arguments.
Step 4	**term ip netmask-format** {**bitcount** | **decimal** | **hexadecimal**} **Example:** `Router(config-line)# ip netmask-format hexadecimal`	Specifies the format the router uses to display the network mask for an individual line.
Step 5	**end** **Example:** `Router(config-if)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Using IP Unnumbered Interfaces on Point-to-Point WAN Interfaces to Limit Number of IP Addresses Required

If you have a limited number of IP network or subnet addresses and you have point-to-point WANs in your network, you can use the IP Unnumbered Interfaces feature to enable IP connectivity on the point-to-point WAN interfaces without actually assigning an IP address to them.

Perform this task to configure the IP Unnumbered Interfaces feature on a point-to-point WAN interface.

IP Unnumbered Feature

The IP Unnumbered Interfaces feature enables IP processing on a point-to-point WAN interface without assigning it an explicit IP address. The IP unnumbered point-to-point WAN interface uses the IP address of another interface to enable IP connectivity, which conserves network addresses.

Note The following restrictions apply to the IP Unnumbered Interfaces feature:

- The IP Unnumbered Interfaces feature is only supported on point-to-point (non-multiaccess) WAN interfaces
- You cannot netboot a Cisco IOS image over an interface that is using the IP Unnumbered Interfaces feature

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no shutdown**
5. **ip address** *ip-address mask*
6. **interface** *type number*
7. **no shutdown**
8. **ip unnumbered** *type number*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 4	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.

	Command or Action	Purpose
Step 5	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 172.16.16.1 255.255.240.0`	Configures the IP address on the interface.
Step 6	**interface** *type number* **Example:** `Router(config-if)# interface serial 0/0`	Specifies a point-to-point WAN interface and enters interface configuration mode.
Step 7	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the point-to-point WAN interface.
Step 8	**ip unnumbered** *type number* **Example:** `Router(config-if)# ip unnumbered fastethernet 0/0`	Enables the IP unnumbered feature on the point-to-point WAN interface. In this example the point-to-point WAN interface uses IP address 172.16.16.1 from Fast Ethernet 0/0.
Step 9	**end** **Example:** `Router(config-if)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

The following commands can help troubleshoot IP addressing:

- **show ip interface** --Displays the IP parameters for the interface.
- **show ip route connected** --Displays the IP networks the networking device is connected to.

Using IP addresses with 31-Bit Prefixes on Point-to-Point WAN Interfaces to Limit Number of IP Addresses Required

You can reduce the number of IP subnets used by networking devices to establish IP connectivity to point-to-point WANs that they are connected to by using IP Addresses with 31-bit Prefixes as defined in RFC 3021.

Perform this task to configure an IP address with a 31-bit prefix on a point-to-point WAN interface.

RFC 3021

Prior to RFC 3021, *Using 31-bit Prefixes on IPv4 Point-to-Point Links* , many network administrators assigned IP address with a 30-bit subnet mask (255.255.255.252) to point-to-point interfaces to conserve IP address space. Although this practice does conserve IP address space compared to assigning IP addresses with shorter subnet masks such as 255.255.255.240, IP addresses with a 30-bit subnet mask still require four addresses per link: two host addresses (one for each host interface on the link), one all-zeros network address, and one all-ones broadcast network address.

The table below shows an example of the four IP addresses that are created when a 30-bit (otherwise known as 255.255.255.252 or /30) subnet mask is applied to the IP address 192.168.100.4. The bits that are used to specify the host IP addresses in bold.

Table 8: Four IP Addresses Created When a 30-Bit Subnet Mask (/30) Is Used

Address	Description	Binary
192.168.100.4/30	All-zeros IP address	11000000.10101000.01100100.00000**00**
192.168.100.5/30	First host addresses	11000000.10101000.01100100.00000**01**
192.168.100.6/30	Second host address	11000000.10101000.01100100.00000**10**
192.168.100.7/30	All-ones broadcast address	11000000.10101000.01100100.00000**11**

Point-to-point links only have two endpoints (hosts) and do not require broadcast support because any packet that is transmitted by one host is always received by the other host. Therefore the all-ones broadcast IP address is not required for a point-to-point interface.

The simplest way to explain RFC 3021 is to say that the use of a 31-bit prefix (created by applying a 31-bit subnet mask to an IP address) allows the all-zeros and all-ones IP addresses to be assigned as host addresses on point-to-point networks. Prior to RFC 3021 the longest prefix in common use on point-to-point links was 30-bits, which meant that the all-zeros and all-ones IP addresses were wasted.

The table below shows an example of the two IP addresses that are created when a 31-bit (otherwise known as 255.255.255.254 or /31) subnet mask is applied to the IP address 192.168.100.4. The bit that is used to specify the host IP addresses in bold

Table 9: Two IP Addresses Created When a 31-Bit Subnet Mask (/31) Is Used

Address	Description	Binary
192.168.100.4/31	First host address	11000000.10101000.01100100.00000100
192.168.100.5/31	Second host address	11000000.10101000.01100100.00000101

The complete text for RFC 3021 is available at http://www.ietf.org/rfc/rfc3021.txt .

Before You Begin

You must have classless IP addressing configured on your networking device before you configure an IP address with a 31-bit prefix on a point-to-point interface. Classless IP addressing is enabled by default in many versions of Cisco IOS software. If you are not certain that your networking device has IP classless addressing configured, enter the **ip classless** command in global configuration mode to enable it.

Note This task can only be performed on point-to-point (nonmultiaccess) WAN interfaces.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip classless**
4. **interface** *type number*
5. **no shutdown**
6. **ip address** *ip-address mask*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip classless** **Example:** `Router(config)# ip classless`	(Optional) Enables IP classless (CIDR). **Note** This command is enabled by default in many versions of Cisco IOS. If you are not certain if it is enabled by default in the version of Cisco IOS that your networking device is running, enter the **ip classless**command as shown. When you are done with this task view the configuration. If the **ip classless** command does not appear in your configuration, it is enabled by default.

	Command or Action	Purpose
Step 4	**interface** *type number* **Example:** `Router(config)# interface serial 0/0`	Specifies a point-to-point WAN interface and enters interface configuration mode.
Step 5	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface.
Step 6	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 192.168.100.4 255.255.255.254`	Configures the 31bit prefix IP address on the point-to-point WAN interface.
Step 7	**end** **Example:** `Router(config-if)# end`	Exits the current configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

The following commands can help troubleshoot IP addressing:

- **show ip interface** --Displays the IP parameters for the interface.
- **show ip route connected** --Displays the IP networks the networking device is connected to.

Configuration Examples for IP Addresses

Example Establishing IP Connectivity to a Network by Assigning an IP Address to an Interface

The following example configures an IP address on three interfaces:

```
!
interface FastEthernet0/0
 no shutdown
 ip address 172.16.16.1 255.255.240.0
!
interface FastEthernet0/1
```

```
 no shutdown
 ip address 172.16.32.1 255.255.240.0
!
interface FastEthernet0/2
 no shutdown
 ip address 172.16.48.1 255.255.240.0
!
```

Example Increasing the Number of IP Hosts that are Supported on a Network by Using Secondary IP Addresses

The following example configures secondary IP addresses on three interfaces:

```
!
interface FastEthernet0/0
 no shutdown
 ip address 172.16.16.1 255.255.240.0
 ip address 172.16.32.1 255.255.240.0 secondary
!
!
interface FastEthernet0/1
 no shutdown
 ip address 172.17.16.1 255.255.240.0
 ip address 172.17.32.1 255.255.240.0 secondary
!
!
interface FastEthernet0/2
 no shutdown
 ip address 172.18.16.1 255.255.240.0
 ip address 172.18.32.1 255.255.240.0 secondary
!
```

Example Using IP Unnumbered Interfaces on Point-to-Point WAN Interfaces to Limit Number of IP Addresses Required

The following example configures the unnumbered IP feature on three interfaces:

```
!
interface FastEthernet0/0
 no shutdown
 ip address 172.16.16.1 255.255.240.0
!
interface serial0/0
 no shutdown
 ip unnumbered fastethernet0/0
!
interface serial0/1
 no shutdown
 ip unnumbered fastethernet0/0
!
interface serial0/2
 no shutdown
 ip unnumbered fastethernet0/0
!
```

Example Using IP addresses with 31-Bit Prefixes on Point-to-Point WAN Interfaces to Limit Number of IP Addresses Required

The following example configures 31-bit prefixes on two interfaces:

```
!
ip classless
!
interface serial0/0
 no shutdown
 ip address 192.168.100.2 255.255.255.254
!
!
interface serial0/1
 no shutdown
 ip address 192.168.100.4 255.255.255.254
```

Example Maximizing the Number of Available IP Subnets by Allowing the Use of IP Subnet Zero

The following example enables subnet zero:

```
!
interface FastEthernet0/0
 no shutdown
 ip address 172.16.16.1 255.255.240.0
!
ip subnet-zero
!
```

Where to Go Next

If your network has two or more routers and you have not already configured a routing protocol, consult the *Cisco IOS IP Routing Protocols Configuration Guide*, Release 12.4T, for information on configuring routing protocols.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IP addressing commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Addressing Services Command Reference*

Related Topic	Document Title
Fundamental principles of IP addressing and IP routing	*IP Routing Primer ISBN 1578701082*

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified	--

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC[6]	Title
RFC 791	*Internet Protocol* http://www.ietf.org/rfc/rfc0791.txt
RFC 1338	*Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy* http://www.ietf.org/rfc/rfc1519.txt
RFC 1466	*Guidelines for Management of IP Address Space* http://www.ietf.org/rfc/rfc1466.txt
RFC 1716	*Towards Requirements for IP Routers* http://www.ietf.org/rfc/rfc1716.txt
RFC 1918	*Address Allocation for Private Internets* http://www.ietf.org/rfc/rfc1918.txt
RFC 3330	*Special-Use IP Addresses* http://www.ietf.org/rfc/rfc3330.txt

6 These references are only a sample of the many RFCs available on subjects related to IP addressing and IP routing. Refer to the IETF RFC site at http://www.ietf.org/rfc.html for a full list of RFCs.

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.htmll

Feature Information for IP Addresses

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 10: Feature Information for IP Addresses

Feature Name	Releases	Feature Information
Classless Inter-Domain Routing	10.0	CIDR is a new way of looking at IP addresses that eliminates the concept of classes (class A, class B, and so on). For example, network 192.213.0.0, which is an illegal class C network number, is a legal supernet when it is represented in CIDR notation as 192.213.0.0/16. The /16 indicates that the subnet mask consists of 16 bits (counting from the left). Therefore, 192.213.0.0/16 is similar to 192.213.0.0 255.255.0.0. The following command was introduced or modified: **ip classless**.

Feature Name	Releases	Feature Information
IP Subnet Zero	10.0	In order to conserve IP address space IP Subnet Zero allows the use of the all-zeros subnet as an IP address on an interface, such as configuring 172.16.0.1/24 on Fast Ethernet 0/0. The following command was introduced or modified: **ip subnet-zero**.
IP Unnumbered Interfaces	10.0	In order to conserve IP address space, IP unnumbered interfaces use the IP address of another interface to enable IP connectivity. The following command was introduced or modified: **ip unnumbered**.
Using 31-bit Prefixes on IP Point-to-Point Links	12.0(14)S 12.2(4)T	In order to conserve IP address space on the Internet, a 31-bit prefix length allows the use of only two IP addresses on a point-to-point link. Previously, customers had to use four IP addresses or unnumbered interfaces for point-to-point links.

CHAPTER 3

IP Overlapping Address Pools

The IP Overlapping Address Pools feature improves flexibility in assigning IP addresses dynamically. This feature allows you to configure overlapping IP address pool groups to create different address spaces and concurrently use the same IP addresses in different address spaces.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for IP Overlapping Address Pools

The Cisco IOS XE software checks for duplicate addresses on a per-group basis. The check for duplicate addresses means that you can configure pools in multiple groups that could have possible duplicate addresses. The IP Overlapping Address Pools feature should be used only in cases where overlapping IP address pools make sense, such as Multiprotocol Label Switching (MPLS) Virtual Private Network (VPN) environments where multiple IP address spaces are supported.

Information About IP Overlapping Address Pools

Benefits

The IP Overlapping Address Pools gives greater flexibility in assigning IP addresses dynamically. It allows you to configure overlapping IP address pool groups to create different address spaces and concurrently use the same IP addresses in different address spaces.

How IP Address Groups Work

IP Control Protocol (IPCP) IP pool processing implements all IP addresses as belonging to a single IP address space, and a given IP address should not be assigned multiple times. IP developments such as virtual private dialup network (VPDN) and Network Address Translation (NAT) implement the concept of multiple IP address spaces where it can be meaningful to reuse IP addresses, although such usage must ensure that these duplicate address are not placed in the same IP address space. An IP address group to support multiple IP address spaces and still allow the verification of nonoverlapping IP address pools within a pool group. Pool names must be unique within the router. The pool name carries an implicit group identifier because that pool name can be associated only with one group. Pools without an explicit group name are considered members of the base system group and are processed in the same manner as the original IP pool implementation.

Existing configurations are not affected by the new pool feature. The “group” concept is an extension of the existing **ip local pool** command. Processing of pools that are not specified as a member of a group is unchanged from the existing implementation.

How to Configure IP Overlapping Address Pools

Configuring and Verifying a Local Pool Group

Perform this task to configure a local pool group and verify that it exists.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip local pool** {**default** | *poolname*} {*low-ip-address* [*high-ip-address*] [**group** *group-name*] [**cache-size** *size*]}
4. **show ip local pool** [*poolname* | [**group** *group-name*]]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip local pool** {**default** | *poolname*} {*low-ip-address* [*high-ip-address*] [**group** *group-name*] [**cache-size** *size*]} **Example:** `Router(config)# ip local pool testpool 10.2.2.1 10.2.2.10 group testgroup cache-size 10000`	Configures a group of local IP address pools, gives this group a name, and specifies a cache size.
Step 4	**show ip local pool** [*poolname* | [**group** *group-name*]] **Example:** `Router(config)# show ip local pool group testgroup testpool`	Displays statistics for any defined IP address pools.

Configuration Examples for Configuring IP Overlapping Address Pools

Define Local Address Pooling as the Global Default Mechanism Example

The following example shows how to configure local pooling as the global default mechanism:

```
ip address-pool local
ip local pool default 192.168.15.15 192.168.15.16
```

Configure Multiple Ranges of IP Addresses into One Pool Example

The following example shows how to configure two ranges of IP addresses for one IP address pool:

```
ip local pool default 192.169.10.10 192.169.10.20
ip local pool default 192.168.50.25 192.168.50.50
```

Additional References

The following sections provide references related to configuring IP Overlapping Address Pools.

Related Documents

Related Topic	Document Title
Dial commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS Dial Services Command Reference*
IP address pooling	"Configuring Media-Independent PPP and Multilink PPP" chapter of the Cisco IOS XE Dial Technologies Configuration Guide

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS XE releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 826	*Address Resolution Protocol*
RFC 903	*Reverse Address Resolution Protocol*
RFC 1027	*Proxy Address Resolution Protocol*
RFC 1042	*Standard for the Transmission of IP Datagrams over IEEE 802 Networks*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for Configuring IP Overlapping Address Pools

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 11: Feature Information for Configuring IP Overlapping Address Pools

Feature Name	Releases	Feature Information
IP Overlapping Address Pools	Cisco IOS XE Release 2.1	The IP Overlapping Address Pools feature improves flexibility in assigning IP addresses dynamically. This feature allows you to configure overlapping IP address pool groups to create different address spaces and concurrently use the same IP addresses in different address spaces. The following commands were modified by this feature:**ip local pool**and **show ip local pool.**

Glossary

IPCP --IP Control Protocol. Protocol that establishes and configures IP over PPP.

MPLS --Multiprotocol Label Switching. Switching method that forwards IP traffic using a label. This label instructs the routers and the switches in the network where to forward the packets based on preestablished IP routing information.

NAT --Network Address Translation. Mechanism for reducing the need for globally unique IP addresses. NAT allows an organization with addresses that are not globally unique to connect to the Internet by translating those addresses into globally routable address space. Also known as Network Address Translator.

VPDN --virtual private dialup network. Also known as virtual private dial network. A VPDN is a network that extends remote access to a private network using a shared infrastructure. VPDNs use Layer 2 tunnel technologies (L2F, L2TP, and PPTP) to extend the Layer 2 and higher parts of the network connection from a remote user across an ISP network to a private network. VPDNs are a cost-effective method of establishing a long distance, point-to-point connection between remote dial users and a private network. See also VPN.

VPN --Virtual Private Network. Enables IP traffic to travel securely over a public TCP/IP network by encrypting all traffic from one network to another. A VPN uses “tunneling” to encrypt all information at the IP level.

VRF --A VPN routing and forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a PE router.

CHAPTER 4

IP Unnumbered Ethernet Polling Support

The IP Unnumbered Ethernet Polling Support feature provides IP unnumbered support for Ethernet physical interfaces. This support already exists for serial interfaces.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About IP Unnumbered Ethernet Polling Support

IP Unnumbered Ethernet Polling Support Overview

IP unnumbered support for serial interfaces is extended to Ethernet physical interfaces. Unnumbered Ethernet physical interfaces are used in the same manner as unnumbered serial interfaces. On a device, if a loopback interface is configured and an IP address is assigned to it, using the polling option more than one Ethernet physical interface can be unnumbered to the loopback.

The polling option enables the dynamic discovery of hosts (connected though the unnumbered interfaces) based on the Address Resolution Protocol (ARP) protocol.

How to Configure IP Unnumbered Ethernet Polling Support

Enabling Polling on an Ethernet Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **exit**
6. **interface** *type number*
7. **ip unnumbered** *type number* **poll**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface loopback 0`	Specifies an interface and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 209.165.200.229 255.255.240.224`	Configures the IP address on the interface.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 6	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 7	**ip unnumbered** *type number* **poll** **Example:** `Device(config-if)# ip unnumbered loopback 0 poll`	Enables IP–connected host polling on the specified interface.
Step 8	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

Configuring the Queue Size and the Packet Rate for IP ARP Polling for Unnumbered Interfaces

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip arp poll queue** *queue-size*
4. **ip arp poll rate** *packet-rate*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip arp poll queue** *queue-size* **Example:** `Device(config)# ip arp poll queue 1000`	Configures the IP ARP polling queue size.
Step 4	**ip arp poll rate** *packet-rate* **Example:** `Device(config)# ip arp poll rate 1000`	Configures the IP ARP polling packet rate, in packets per second.
Step 5	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

Verifying IP Unnumbered Ethernet Polling Support

Perform this task to verify IP unnumbered Ethernet polling support.

Note The **show** commands are not in any specific order.

SUMMARY STEPS

1. **enable**
2. **show ip arp poll**
3. **show ip interface** *type number* **unnumbered**
4. **show ip interface** *type number* **unnumbered detail**

DETAILED STEPS

Step 1 **enable**
Enables privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 **show ip arp poll**

Displays the IP ARP host polling status.

Example:

```
Device# show ip arp poll

Number of IP addresses processed for polling: 438
Number of entries in the queue: 100 (high water mark: 154, max: 1000)
Number of request dropped:
  Queue was full: 1288
  Request was throttled by incomplete ARP: 10
  Duplicate entry found in queue: 1431
```

Step 3 **show ip interface** *type number* **unnumbered**

Displays the status of unnumbered interface support on interfaces configured for IP.

Example:

```
Device# show ip interface loopback 0 unnumbered

Number of unnumbered interfaces with polling: 10
Number of IP addresses processed for polling: 15
Number of IP addresses in queue for polling: 4
```

Step 4 **show ip interface** *type number* **unnumbered detail**

Displays the detailed status of unnumbered interface support on interfaces configured for IP.

Example:

```
Device# show ip interface loopback 0 unnumbered detail

Number of unnumbered interfaces with polling: 10
Number of IP addresses processed for polling: 15
Last 10 IP addresses processed for polling:
  209.165.201.2
  209.165.201.3
  209.165.201.4
  209.165.201.5
  209.165.201.6
  209.165.201.7
  209.165.201.8
  209.165.201.9
  209.165.201.10
  209.165.201.11
Number of IP addresses in queue for polling: 4 (high water mark: 5)
  209.165.201.12
  209.165.201.13
  209.165.201.14
  209.165.201.15
```

Configuration Examples for IP Unnumbered Ethernet Polling Support

Example: Enabling Polling on an Ethernet Interface

```
Device> enable
Device# configure terminal
Device(config)# interface loopback 0
Device(config-if)# ip address 209.165.200.229 255.255.240.224
Device(config-if)# exit
Device(config)# interface ethernet 0/0
Device(config-if)# ip unnumbered loopback 0 poll
Device(config-if)# end
```

Example: Configuring the Queue Size and the Packet Rate for IP ARP Polling for Unnumbered Interfaces

```
Device> enable
Device# configure terminal
Device(config)# ip arp poll queue 1000
Device(config)# ip arp poll rate 1000
Device(config)# end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
IPv4 Addressing commands	Cisco IOS IP Addressing Services Command Reference
Conceptual information about IPv4 addresses	"Configuring IPv4 Addresses" module in the *IP Addressing: IPv4 Addressing Configuration Guide*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IP Unnumbered Ethernet Polling Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 12: Feature Information for IP Unnumbered Ethernet Polling Support

Feature Name	Releases	Feature Information
IP Unnumbered Ethernet Polling Support	Cisco IOS XE Release 3.8S	The IP Unnumbered Ethernet Polling Support feature provides IP unnumbered support for Ethernet physical interfaces. The following commands were introduced or modified: **clear ip arp poll statistics**, **clear ip interface**, **ip arp poll**, **ip unnumbered poll**, **show ip arp poll**, and **show ip interface unnumbered**.

CHAPTER 5

Auto-IP

The auto-IP feature automatically provides IP addresses to the nodes inserted into a ring. In ring topology, when a device is inserted into the ring, the neighboring node interfaces require manual reconfiguration. The auto-IP feature addresses the problem of manually reconfiguring nodes during insertion, deletion, and movement of nodes within the ring. The auto-IP feature is supported on the following:

- Ethernet interfaces and sub interfaces.
- Virtual routing and forwarding instance (VRF) interfaces.
- Switch Virtual Interfaces (SVIs).
- EtherChannels.

Attention To know the release versions that support the auto-IP feature on VRF interfaces, SVIs, and EtherChannels, refer Feature Information for Auto-IP.

Note When a device is inserted into a ring, it is called a node.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Auto-IP

- Link Layer Discovery Protocol (LLDP) must be enabled on the device before the auto-IP functionality is enabled on the node interface.

Auto-IP on an EtherChannel

- When you configure auto-IP on an EtherChannel, ensure that LLDP is enabled on the member interfaces of the EtherChannel.
- Auto-IP configuration on an interface must be removed before moving an interface into an EtherChannel.

Auto-IP on VRF interfaces

- If you intend to configure auto-IP on an interface for a specific virtual routing and forwarding instance (VRF), then ensure that the interface is presently within the VRF. If you enable auto-IP on an interface and then associate the interface to a VRF, the auto-IP settings on the interface will be cleared, and you will have to enable the auto-IP feature on the VRF interface again.

Restrictions for Auto-IP

- Auto-IP addresses must not contain an even number in the last octet (such as 10.1.1.2, where the number in the last octet is 2).

Auto-IP on VRF interfaces

- Auto-IP configuration on an interface is not retained when the interface is moved from one virtual routing and forwarding instance (VRF) to another, including the global VRF.
- Interface nodes in different VRFs cannot be configured for the same ring. Ensure that the nodes you select belong to the same VRF.
- If a VRF address family is IPv6, you cannot configure auto-IP on the interfaces within the VRF. You can configure auto-IP on a VRF interface if the VRF address family is IPv4.

Auto-IP on SVI interfaces

- Auto-IP configuration is not possible on a Switch Virtual Interface (SVI) with more than one physical interface. The SVI physical interface must be an access port or trunk port with only one associated VLAN or a bridge domain interface (BDI).

Auto-IP on EtherChannel interfaces

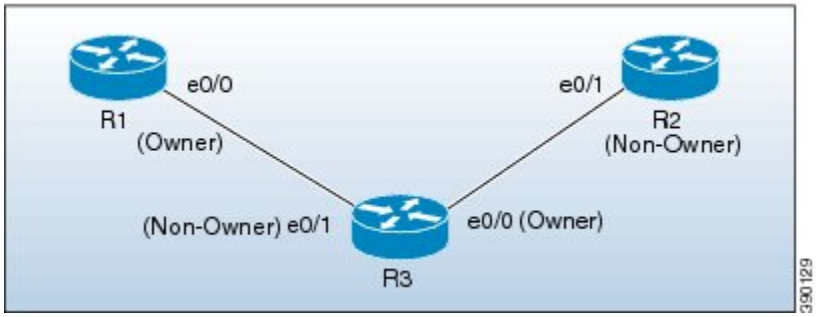
- Auto-IP configuration can be done on an EtherChannel interface, but not on a member interface of the EtherChannel.

Information About Auto-IP

Auto-IP Overview

The auto-IP feature is an enhancement of Link Layer Discovery Protocol (LLDP). LLDP uses a set of attributes to discover neighbor devices. This attribute set is called Type Length Value (TLV) as it contains type, length, and value descriptions.

In a ring topology, two network-to-network interfaces (NNIs or node interfaces) of a device are used to be part of the ring. For a ring to function as an auto-IP ring, you must configure the auto-IP feature on all the node interfaces within the ring. One node interface of a device is designated as the owner-interface and the other interface as the non-owner-interface. In an auto-IP ring, the owner-interface of a device is connected to a non-owner-interface of the neighbor device. A sample topology is given below:



When a new device is inserted into an auto-IP ring, owner and non-owner-interfaces of the inserted device are identified. The node interface of the inserted device that is connected to an owner-interface is designated as the non-owner-interface, and it automatically receives an IP address from the connected neighbor device. The IP address is automatically configured on the interface. Since the non-owner-interface is identified, the other node interface of the inserted device is designated as the owner-interface, and the device assigns a pre configured auto-IP address to its designated owner-interface.

An auto-IP address is a preconfigured address configured on a node interface to make the interface capable of automatically assigning an IP address to a new neighbor interface that is detected in the auto-IP ring. The configured auto-IP address is used for allocation purposes.

You must configure the same auto-IP address on the two node interfaces that are designated to be part of an auto-IP ring, and the auto-IP address must contain an odd number in the last octet. The auto-IP address is assigned to the owner-interface when the device is introduced into an auto-IP ring. Since each auto-IP address contains an odd number in the last octet, the IP address derived by subtracting 1 from the last octet is an even number, and is not used for designating auto-IP addresses. This IP address is allocated to a newly detected neighbor, non-owner-interface.

For example, if we assume that the device R3 is inserted between the devices R1 and R2 in the above topology, and the auto-IP address 10.1.1.3 is configured on e0/1 and e0/0, the two node interfaces on device R3, then R1 assigns an IP address to the non-owner-interface of R3, e0/1. The IP address 10.1.1.3 is assigned to the

owner-interface of R3, e0/0. The IP address derived by subtracting 1 from the last octet of the auto-IP address is 10.1.1.2. 10.1.1.2 is assigned to the neighbor non-owner-interface of the connected neighbor device R2.

Auto-IP TLV exchange

Before insertion, the node interfaces are not designated as owner and non-owner. After insertion, the auto-IP TLV is exchanged between the neighbor devices. During this initial negotiation with the adjacent device interfaces, owner and non-owner-interfaces are determined automatically.

After a device is inserted into a ring, the auto-IP address configured for the device (such as 10.1.1.3) is assigned to the owner-interface for the /31 subnet. An owner-interface has a priority 2 in the auto-IP TLV, and a non-owner-interface has priority 0 in the auto-IP TLV. If there is no assigned IP address on the node interface (before the node is inserted into a ring), then the ring interface has priority 1 in the auto-IP TLV.

The IP address negotiation is based on priority; the higher value of priority wins the negotiation. If the priority is equal, then IP negotiation fails. This scenario usually occurs when there is an incorrect configuration or wiring. In such a scenario, you must ensure that the configuration and wiring is proper.

Auto-IP on VRF interfaces

Some points on auto-IP configuration on virtual routing and forwarding instance (VRF) interfaces are noted below:

- Auto-IP configuration on an interface is removed when the interface is moved from one VRF to another, including the global VRF. So, assign the interface to a VRF and then configure the auto-IP feature on the interface.
- You can configure auto-IP on a VRF interface only if the address family of the VRF is IPv4. If the IPv4 address family configuration is removed from a VRF, the auto-IP configuration is removed from all the interfaces within the VRF.
- If a VRF address family is IPv6, you cannot configure auto-IP on the interfaces within the VRF. However, if a VRF address family is IPv4 and IPv6, you can configure auto-IP on the interfaces within the VRF.
- If the IPv6 address family configuration is removed from a VRF with both IPv4 and IPv6 address-family configuration, the auto-IP configuration on the interfaces within the VRF remain intact.
- If a VRF is deleted, then the auto-IP configuration on all the interfaces assigned to the VRF are removed.
- A specific ring has two interface nodes. Ensure that the two nodes you select belong to the same VRF. Nodes in different VRFs cannot be configured for the same ring.
- Within a VRF, the same auto-IP address cannot be used for different ring IDs.

Auto-IP on EtherChannel interfaces

Some points on auto-IP configuration for an EtherChannel interface are noted below:

- You can configure auto-IP on an EtherChannel interface. If you configure the auto-IP feature on an EtherChannel and then add member interfaces to the EtherChannel, then auto-IP TLV information is carried to all the member interfaces. If you add member interfaces to the EtherChannel and then configure auto-IP on the EtherChannel, auto-IP TLV information is carried to all the member interfaces.

Attention LLDP must be enabled on the member interfaces.

- The list of EtherChannel member interfaces are maintained in ring interfaces corresponding to the EtherChannel. Auto-IP information is transmitted on all the EtherChannel member interfaces.
- If you remove a member interface from an EtherChannel, auto-IP TLV information is not carried to the removed interface.

Auto-IP on SVI interfaces

Some points on auto-IP configuration on a Switch Virtual Interface (SVI) are noted below:

- Auto-IP configuration on an SVI is possible only if a single physical interface is associated with an SVI.
- The SVI physical interface must be an access port or trunk port with only one associated VLAN or a bridge domain interface (BDI).
- If the SVI is mapped to more than one physical port, then the auto-IP configuration on the SVI will be removed.

Seed Device

Seed devices are the devices used to initiate network discovery. To initiate auto-IP capability in a ring, at least one device must be configured as a seed device in the ring. To configure a device as a seed device in an auto-IP ring, you must manually configure the IP address configured on one of its node interfaces with the auto-IP address of the interface, with the mask /31 (or 255.255.255.254).

A sample topology is given below. In this scenario, device R1 is being configured as the seed device.

The e0/0 interface on device R1 is configured with the auto-IP address 10.1.1.1 and the e0/1 interface on device R2 is configured with the auto-IP address 10.1.1.3.

To configure R1 as the seed device, 10.1.1.1 must be configured as the IP address of the interface e0/0. By configuring the IP address of e0/0 interface of R1 to its auto-IP address, R1 is configured as the seed device and the interface e0/0 becomes the owner of the subnet.

The process of configuring the device R1 as the seed device is given below:

After a connection is established between the devices R1 and R2, R1 sends a Link Layer Discovery Protocol(LLDP) packet which contains an auto-IP Type Length Value (TLV) with priority 2.

The auto-IP information for the e0/0 interface on R1 is given below:

Interface IP address	Auto-IP address	Priority
10.1.1.1	10.1.1.1	2

On receiving the auto-IP TLV from R1, R2 derives the IP address for the interface e0/1 (by subtracting 1 from the last octet of R1's auto-IP address), and assigns the IP address 10.1.1.0/31 to R2's e0/1 interface. The interface e0/1 on R2 becomes the non-owner interface on this subnet.

The IP address allocation is displayed in the illustration given below:

The device and node interface details for the subnet are given below:

Device	Interface	IP address	Designation
R1	e0/0	10.1.1.1/31	Owner
R2	e0/1	10.1.1.0/31	Non-owner

Note Since the auto-IP address configured on the e0/1 interface on R2 is 10.1.1.3, the other node interface of R2 is designated as the owner interface and 10.1.1.3 is automatically configured as the interface IP address of the other node interface.

Auto-IP Configuration for Inserting a Device into an Auto-IP Ring

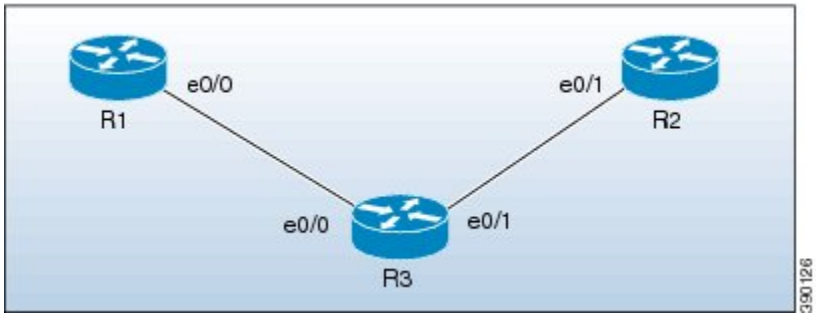
To insert a device into an existing auto-IP ring, the node interfaces of the device must be configured with the auto-IP address.

Note You can also configure the auto-IP feature on node interfaces that are part of an existing, but non-auto-IP ring.

The topology in the illustration below shows a sample scenario.

Device R1 is configured as the seed device. Interface e0/0 on R1 is configured with the IP address 10.1.1.1/31, and is the owner of the subnet connecting R1 and R2. Interface e0/1 on device R2 has the IP address 10.1.1.0/31, and is the non-owner interface of the subnet.

Device R3 is inserted between R1 and R2. The two designated node interfaces e0/0 and e0/1 of R3 are configured with the auto-IP address 10.1.1.5. After insertion of the device, the ring topology appears as shown in the illustration below:



Auto-IP TLV exchange between the devices R1 and R3 is given below:

1. R1 sends an auto-IP Type Length Value (TLV) with priority 2 to the e0/0 interface of R3.
2. After receiving the auto-IP TLV from R1, R3 sends an auto-IP TLV with priority 0 to the e0/0 interface of R1.
3. R1 wins the election process and the interface e0/0 of R1 is designated as the owner interface on the subnet connecting R1 and R3.
4. The e0/0 interface on R3 becomes the non-owner interface and the IP address 10.1.1.0 is assigned to it.
5. The other node interface on R3 is designated as an owner interface and its auto-IP address (10.1.1.5) is assigned as the IP address of the interface.

Auto-IP TLV exchange between the devices R3 and R2 is given below:

1. R3 sends an auto-IP TLV with priority 2 to the e0/1 interface of R2.
2. After receiving the auto-IP TLV from R3, R2 sends an auto-IP TLV with priority 0 to the e0/1 interface of R3.
3. R3 wins the election process and its interface e0/1 is designated as the owner interface on the subnet connecting R3 and R2.
4. The e0/1 interface on R2 is designated as the non-owner interface, and the IP address 10.1.1.4 is assigned to it.
5. The other node interface on R2 is designated as the owner interface and its auto-IP address is assigned as the IP address.

The IP addresses that are configured for the owner and non-owner interfaces on the devices R1, R2, and R3 are given below:

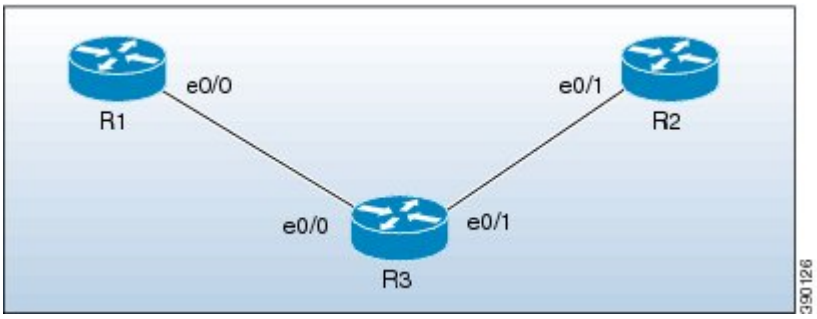
Device	Interface	IP Address	Designation
R1	e0/0	10.1.1.1/31	Owner
R3	e0/0	10.1.1.0/31	Non-owner
R3	e0/1	10.1.1.5/31	Owner
R2	e0/1	10.1.1.4/31	Non-owner

Device Removal from an Auto-IP Ring

You can manually remove a device from an existing auto-IP ring.

Note No configuration is required if you remove a device from an auto-IP ring and connect its neighbor devices.

The topology in the illustration below shows a sample scenario:



In the topology, device R3 is removed from the auto-IP ring and device R1 is connected to R2. As a result, auto-IP Type Length Value (TLVs) are exchanged between R1 and R2. Since the e0/0 interface of R1 sends an auto-IP TLV with priority 2 and the e0/1 interface of R2 sends an auto-IP TLV with priority 0 to the e0/0 interface on R1, the e0/0 interface of R1 is designated as the owner interface on the subnet connecting R1 and R2. R1 assigns the IP address to the e0/1 interface on R2, and it becomes the non-owner interface on this subnet.

After the removal of R3 from the auto-IP ring, the ring topology looks like this:

The IP address of the owner and non-owner interfaces on the subnet are given below:

Device	Interface	Designation
R1	e0/0	Owner
R2	e0/1	Non-owner

Conflict Resolution Using the Auto-Swap Technique

The auto-swap technique automatically resolves conflicts due to incorrect insertion of a device into an auto-IP ring.

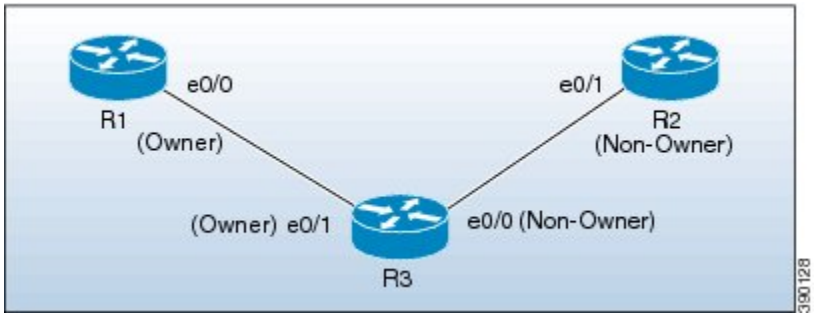
If you remove a device from an auto-IP ring, the owner and non-owner auto-IP configuration on the node interfaces is retained. You can insert the device back into an auto-IP ring.

If you incorrectly insert a device into a ring with its interfaces swapped (due to which two owner interfaces and two non-owner interfaces are connected to each other, rather than a connection between an owner and a non-owner interface), then identical priority values are exchanged between interfaces during the auto-IP Type Length Value (TLV) transmission. This leads to a tie in the priority value that is exchanged between the node interfaces of the inserted device, and a conflict is detected.

The auto-swap technique resolves conflicts on both the node interfaces of the inserted device and allows allocation of IP addresses for the interfaces.

Note No configuration is required to enable the auto-swap technique; it is enabled automatically. The auto-swap technique is used only when conflict is detected on both the node interfaces of the device.

The topology in the illustration below shows a sample scenario:



In this topology, device R3 is incorrectly inserted between the devices R1 and R2, with its interfaces swapped. The conflict arises due to incorrect insertion, as given below:

- An owner interface is connected to another owner interface; the e0/0 interface of R1 is connected to the e0/1 interface of R3.
- A non-owner interface is connected to another non-owner interface; the e0/1 interface of R2 is connected to the e0/0 interface of R3.

The auto-IP TLV exchange details between R1 and R3 are given below:

- The e0/0 interface on R1 sends an auto-IP TLV with priority 2 to the e0/1 interface on R3.
- The e0/1 interface on R3 sends an auto-IP TLV with priority 2 to the e0/0 interface on R1.

Since the same priority value of 2 is sent in both instances, there is a tie during the election process, leading to a conflict.

Similarly, the same priority value of 0 is exchanged between the e0/0 interface of R3 and the e0/1 interface of R2 since they are non-owner interfaces, leading to a conflict.

Auto Swap

The auto-IP feature uses the auto-swap technique to resolve conflicts on both the node interfaces of the inserted device.

The priority and the interface IP address of the e0/1 interface on R3 is swapped with the priority and the interface IP address of the e0/0 interface on R3, respectively.

After swapping, the following auto-IP TLV information is exchanged between R1 and R3:

- The e0/0 interface on R1 sends an auto-IP TLV with priority 2 to the e0/1 interface on R3.
- The e0/1 interface on R3 sends an auto-IP TLV with priority 0 to the e0/0 interface on R1.

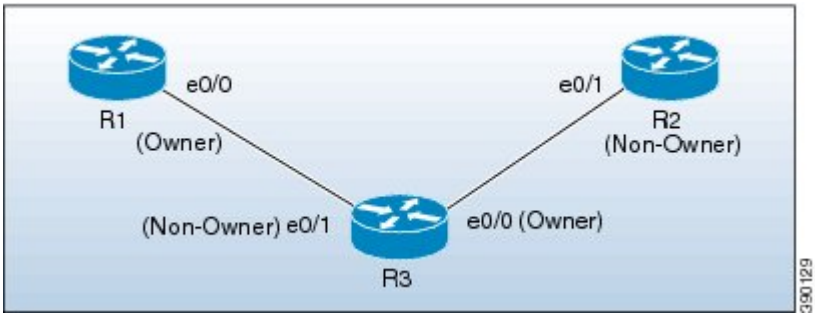
Since the priority sent by R1 to R3 is higher than the priority sent by the interface e0/1 on R3, R3 derives the IP address 10.1.1.0 for the e0/1 interface from the auto-IP address of R1 (10.1.1.1).

The following auto-IP TLV information is exchanged between R3 and R2:

- The e0/0 interface on R3 sends an auto-IP TLV with priority 2 to the e0/1 interface on R2.
- The e0/1 interface on R2 sends an auto-IP TLV with priority 0 to the e0/1 interface on R3.

R2 detects the priority sent by R3 to be higher than the priority sent by its interface e0/1 and derives the IP address 10.1.1.4 from the auto-IP address of R3 (10.1.1.5).

After conflict resolution, the topology looks like this:



The e0/1 interface on R3 is designated as a non-owner interface and the e0/0 interface on R3 is designated as the owner interface.

How to Configure Auto-IP

Configuring a Seed Device

You must configure at least one seed device in an auto-IP ring. To configure a seed device, you must configure the auto-IP address on the two node interfaces of the device (for a specific ring), and use the same IP address to configure the IP address on one of the two node interfaces.

Attention Understand these concepts before configuring auto-IP on virtual routing and forwarding instance (VRF) interfaces, Switch Virtual Interfaces (SVIs), and EtherChannels:

- VRF—If you intend to enable auto-IP on a VRF interface, ensure that the node interface is presently within the VRF. If the interface is not within a VRF presently, assign the interface to the VRF and then configure auto-IP on the VRF interface. Ensure that both node interfaces for the ring are assigned to the same VRF.
- SVI—Auto-IP configuration on an SVI is possible only if a single physical interface is associated with an SVI and the physical interface is an access port.
- EtherChannels—You can configure auto-IP on an EtherChannel interface, but not on a member interface of the EtherChannel.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp run**
4. **interface** *type number*
5. **auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address*
6. **exit**
7. **interface** *type number*
8. **auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address*
9. **ip address** *interface-ip-address subnet-mask*
10. **end**
11. **show auto-ip-ring** [*ring-id*][**detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**lldp run** **Example:** `Device(config)# lldp run`	Enables Link Layer Discovery Protocol (LLDP) for the device.
Step 4	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/0`	Specifies an interface type and number, and enters interface configuration mode.
Step 5	**auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address* **Example:** `Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.1`	Configures the auto-IP address on the specified interface.
Step 6	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/1`	Specifies an interface type and number, and enters interface configuration mode.
Step 8	**auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address* **Example:** `Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.1`	Configures the auto-IP address on the specified interface.
Step 9	**ip address** *interface-ip-address subnet-mask* **Example:** `Device(config-if)# ip address 10.1.1.1 255.255.255.254`	Configures the IP address on the specified interface. **Note** The specified interface is designated as the owner interface of the seed device.
Step 10	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 11	**show auto-ip-ring** [*ring-id*][**detail**] **Example:** `Device# show auto-ip-ring 4 detail`	Displays auto-IP information.

Configuring the Auto-IP Functionality on Node Interfaces (for Inclusion in an Auto-IP Ring)

To insert a device into an auto-IP ring or to enable node interfaces in an existing ring, you must configure the auto-IP address on the 2 designated node interfaces of the device.

Attention Understand these concepts before configuring auto-IP on virtual routing and forwarding instance (VRF) interfaces, Switch Virtual Interfaces (SVIs), and EtherChannels:

- VRF—If you intend to enable auto-IP on a VRF interface, ensure that the node interface is presently within the VRF. If the interface is not within a VRF presently and you want the interface to be within a VRF, move the interface within the VRF and then configure auto-IP on the VRF interface. Ensure that both node interfaces are within the same VRF.
- SVI—Auto-IP configuration on an SVI is possible only if a single physical interface is associated with an SVI and the physical interface is an access port.
- EtherChannels—You can configure auto-IP on an EtherChannel interface, but not on a member interface of the EtherChannel.

This task is applicable for a non-seed device in an auto-IP ring. Ensure that a seed device is configured for the auto-IP ring before performing this task.

Perform the steps given below to configure the auto-IP functionality on the two node interfaces of a device:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp run**
4. **interface** *type number*
5. **auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address*
6. **exit**
7. **interface** *type number*
8. **auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address*
9. **end**
10. **show auto-ip-ring** [*ring-id*][**detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**lldp run** **Example:** `Device(config)# lldp run`	Enables Link Layer Discovery Protocol (LLDP) for the device.
Step 4	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/1`	Specifies an interface type and number, and enters interface configuration mode.
Step 5	**auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address* **Example:** `Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.3`	Configures the auto-IP address on the specified interface.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface ethernet 1/1`	Specifies an interface type and number, and enters interface configuration mode.
Step 8	**auto-ip-ring** *ring-id* **ipv4-address** *auto-ip-address* **Example:** `Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.3`	Configures the auto-IP address on the specified interface.
Step 9	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 10	**show auto-ip-ring** [*ring-id*][**detail**] **Example:** `Device# show auto-ip-ring 4 detail`	Displays auto-IP information.

Verifying and Troubleshooting Auto-IP

Perform this task to verify auto-IP functions.

Note The commands are not in any specific order. The **show auto-ip-ring** command is presented twice. One of the examples displays auto-IP ring information for virtual routing and forwarding instance (VRF) interfaces, and the other example displays auto-IP ring information for non-VRF interfaces.

SUMMARY STEPS

1. **enable**
2. **show auto-ip-ring** [*ring-id*][**detail**]
3. **show auto-ip-ring** [*ring-id*][**detail**]
4. **debug auto-ip-ring** {*ring-id* {**errors** | **events**} |**errors** | **events**}

DETAILED STEPS

Step 1 **enable**
Enables privileged EXEC mode.

Example:
```
Device> enable
```

Step 2 **show auto-ip-ring** [*ring-id*][**detail**]
This command displays auto-IP ring information for a specific device or auto-IP ring.

Example:

```
Device# show auto-ip-ring

Auto-IP ring 1
Auto-IP Address         : 10.1.1.5

Ring Port0              : Ethernet0/0
My Current-IP           : 0.0.0.0
My Priority             : 1

Auto-IP ring 3
Auto-IP Address         : 10.1.1.3

Ring Port0              : Ethernet0/1
My Current-IP           : 0.0.0.0
My Priority             : 1
```

Step 3 **show auto-ip-ring** [*ring-id*][**detail**]
This command displays auto-IP ring information for VRF interfaces.

Example:

```
Device# show auto-ip-ring detail

Auto-IP ring 7
Auto-IP Address         : 10.1.1.11

VRF Name                : 3
Ring Port1              : Ethernet1/1
My Current-IP           : 10.1.1.11
My Priority             : 2

Rx Auto-IP Address      : 10.1.1.13
Rx Current-IP           : 10.1.1.10
Rx Priority             : 0


VRF Name                : 3
Ring Port0              : Ethernet1/0
My Current-IP           : 10.1.1.8
My Priority             : 0

Rx Auto-IP Address      : 10.1.1.9
Rx Current-IP           : 10.1.1.9
Rx Priority             : 2
```

Step 4 **debug auto-ip-ring** {*ring-id* {**errors** | **events**} |**errors** | **events**}
This command debugs errors and events for the specified auto-IP ring.

Example:

```
Device# debug auto-ip-ring 1 errors

Auto IP Ring errors debugging is on for the ring id : 1
*Jul 26 11:30:40.541: (Ethernet0/0) priority (value:1) conflict detected,  need admin intervention
```

Note A conflict is detected in the above debug example because the priority in the auto-IP Type Length Value (TLV) that is sent from the interface and the priority that is received from the neighbor interface is the same.

Configuration Examples for Auto-IP

Example: Configuring a Seed Device

```
Device> enable
Device# configure terminal
Device(config)# lldp run
Device(config)# interface ethernet 0/0
Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.1
Device(config-if)# exit
Device(config)# interface ethernet 1/0
Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.1
Device(config-if)# ip address 10.1.1.1 255.255.255.254
Device(config-if)# end
```

Example: Configuring the Auto-IP Functionality on Node Interfaces (for Inclusion in an Auto-IP Ring)

```
Device> enable
Device# configure terminal
Device(config)# lldp run
Device(config)# interface ethernet 0/1
Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.3
Device(config-if)# exit
Device(config)# interface ethernet 1/1
Device(config-if)# auto-ip-ring 4 ipv4-address 10.1.1.3
Device(config-if)# end
```

Additional References for Auto-IP

Related Documents

Related Topic	Document Title
Configuring IPv4 Addresses	IP Addressing: IPv4 Addressing Configuration Guide
Using Link Layer Discovery Protocol in Multivendor Networks	Carrier Ethernet Configuration Guide
IPv4 Addressing commands	Cisco IOS IP Addressing Services Command Reference
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Auto-IP

Table 13: Feature Information for Auto-IP

Feature Name	Releases	Feature Information
Auto-IP		The auto-IP feature addresses the problem of manually reconfiguring nodes during insertion, deletion, and movement of nodes within an auto-IP ring. The auto-IP feature automatically provides IP addresses to the node interfaces inserted into an auto-IP ring. The following commands were introduced or modified: **auto-ip-ring**, **debug auto-ip-ring**, **show auto-ip-ring**.
		The following commands were introduced or modified: **show auto-ip-ring**.

CHAPTER 6

Zero Touch Auto-IP

The Zero touch Auto-IP feature enables automatic allocation and configuration of IP addresses for nodes in a ring topology. The IP addresses are allocated from a pool of IP addresses that is predefined by you.

The advantages of Zero Touch Auto-IP over Auto-IP are:

- IP addresses can be configured automatically on ring nodes. Manual IP address configuration is not required on each node.
- IP addresses are allocated from a common IP address pool, and the IP address range can be predefined by you.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Zero Touch Auto-IP

- Link Layer Discovery Protocol (LLDP) must be enabled on all the Auto-IP ring device ports.
- In an Auto-IP ring, you must identify one Auto-IP device as an Auto-IP server.
- None of the ports identified to be part of the Zero Touch Auto-IP ring should be manually configured with the Auto-IP functionality. If a port that is identified for Zero touch Auto-IP configuration has a manual Auto-IP configuration, disable the manual Auto-IP configuration on that port.

Restrictions for Zero Touch Auto-IP

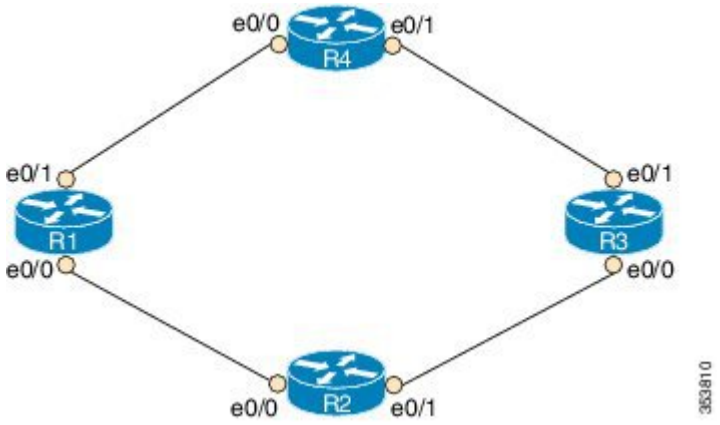
- Zero Touch Auto-IP and Auto-IP cannot coexist. To implement Zero Touch Auto-IP functionality, all the ports of the Auto-IP ring have to be configured as Zero Touch Auto-IP ports.
- Zero Touch Auto-IP works if the designated Auto-IP server is in an autonomic network.

Information About Zero Touch Auto-IP

The Zero Touch Auto-IP feature uses Autonomic Networking and Link Layer Discovery Protocol (LLDP) to achieve the objective of automatic IP address configuration on nodes in a ring network.

Consider the topology for Zero Touch Auto-IP configuration. The devices R1, R2, R3 and R4 are connected in a ring network and LLDP is enabled on all the ring ports.

Figure 7: Zero Touch Auto-IP Topology



To know about and configure the Zero touch Auto-IP functionality, use the information given below:

1 Associate one device in the ring network (say R1) to the autonomic network. Enable autonomic status for the other Auto-IP devices. For more information on autonomic networks, refer Autonomic Networking.

```
R1(config)# autonomic registrar
R1(config-registrar)# domain-id auto-addressing.com
R1(config-registrar)# no shutdown
```

```
R1(config-registrar)# CA local
R1(config-registrar)# exit
R1(config)# autonomic


R2(config)# autonomic
R3(config)# autonomic
R4(config)# autonomic
```

Note that R1 is configured on the registrar and receives a certificate. The remaining devices are configured as autonomic devices.

2 Enable the *auto* mode on all the ports in the ring to enable automatic IP address configuration. Auto mode must be enabled on the e0/0 and e0/1 ports on R1, R2, R3 and R4. For ports of the same device, the ring ID must be identical.

```
Device(config-if)# auto-ip-ring 1 ipv4-auto
```

3 Configure the device added to the autonomic network (R1) as the Auto-IP server. The server stores a pool of IP addresses.

```
R1(config)# auto-ip-ring server
```

4 Reserve a pool of IP addresses on the Auto-IP server for IP address allocation to the ring ports.

Note In Zero Touch and manual Auto-IP configuration, a /31 subnet is created for a pair of owner and nonowner ports (each device will have a owner and non owner port). An odd-numbered IP address (such as 10.1.1.11) is issued to an owner port and an even-numbered IP address (10.1.1.10) is reserved for a nonowner port. Therefore, specify the first IP address in the range along with the number of devices (or /31 subnets) that make up the Auto-IP ring

```
R1(config-auto-ip-server)# ipv4-address-pool 10.1.1.10 6
```

Result—A range of IP addresses from 10.1.1.10 to 10.1.1.21 is allocated for the Auto-IP ring. The Auto-IP server is added to the autonomic network and is reachable by other nodes in the autonomic network.

Note IP addresses for six devices will be reserved (though the requirement is for four devices); the additional IP addresses will be allocated when you add new devices to the ring.

5 Auto-IP negotiation process— IP addresses are allocated to the Auto-IP ring nodes through a negotiation process. To initiate the process, configure one port as the seed port in the Auto-IP ring.

```
R1(config-if)# auto-ip-ring 1 ipv4-seed
```

The negotiation process is explained below:

1 The priority of the seed port (a port on R1, for example) is set to 2 and it is made an owner port. An IP address from the reserved pool is configured on the port.

2 The seed port advertises its priority (2) to its connected neighbor, and makes the neighbor port a non owner. The seed port assigns an IP address to the neighbor port and the neighbor port's priority is changed to 0.

3 Each owner port in the ring gets an IP address from the Auto IP server. The owner port, in turn, assigns an IP address to the connected neighbor port.

6 Auto-IP communication—After initial configuration, each owner port sends periodic messages to the Auto-IP server to continue preserving its IP address. If there is no message from the owner port to the Auto-IP server for 15 minutes, the server moves the IP address to the pool of free IP addresses.

The following are some points to keep in mind in the context of Zero Touch Auto-IP configuration:

- LLDP has to be enabled on all the Auto-IP ring ports before Auto-IP configuration.
- Before you insert a new interface into the ring, configure auto mode on the ring ports.
- For Zero Touch Auto-IP configuration, the number of devices (or /31 subnets) that make up the Auto-IP ring must be between 1 and 128.
- When you specify a pool of IP addresses, ensure that IP addresses in the specified range are not already in use.
- Ensure that you reserve some additional IP addresses for the Auto-IP ring, in case more devices are added to the ring topology at a later point in time.
- The starting IP address used for the Auto-IP address pool reservation must be an even number. For example, 10.1.1.10 is a valid IP address but 10.1.1.9 is not.
- If you remove a device from an Auto-IP ring, the Auto-IP addresses are released back to the Auto-IP server.

How to Configure Zero Touch Auto-IP

Associating an Auto-IP Server with an Autonomic Network

The Auto-IP server (R1) must be associated with the autonomic network, and configured in the Autonomic Network registrar. The other devices in the network (R2, R3, and R4) must be enabled with the autonomic status.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **autonomic registrar**
4. **domain-id auto-addressing.com**
5. **no shutdown**
6. **CA local**
7. **exit**
8. **autonomic**
9. **autonomic**
10. **autonomic**
11. **autonomic**
12. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `R1> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `R1# configure terminal`	Enters global configuration mode.
Step 3	**autonomic registrar** **Example:** `R1(config)# autonomic registrar`	Enables the Auto-IP server in the Autonomic Network registrar and enters registrar configuration mode.
Step 4	**domain-id auto-addressing.com** **Example:** `R1(config-registrar)# domain-id auto-addressing.com`	Represents a common group of all devices registering with the registrar. **Note** If R1 is configured on the AN registrar, then R1 represents the Auto-IP ring devices R2, R3, and R4.
Step 5	**no shutdown** **Example:** `R1(config-registrar)# no shutdown`	Enables the autonomic registrar.
Step 6	**CA local** **Example:** `R1(config-registrar)# CA local`	Issues a Local CA certificate to the Auto-IP server.
Step 7	**exit** **Example:** `R1(config-registrar)# exit`	Exits registrar configuration mode and enters global configuration mode.
Step 8	**autonomic** **Example:** `R1(config)# autonomic`	Configures the Auto-IP server as an autonomic device. **Note** You should associate the remaining devices (R2, R3, and R4) in the Auto-IP ring with the autonomic network, as given in the next few steps.

	Command or Action	Purpose
Step 9	**autonomic** **Example:** `R2(config)# autonomic`	Configures R2 as an autonomic device.
Step 10	**autonomic** **Example:** `R3(config)# autonomic`	Configures R3 as an autonomic device.
Step 11	**autonomic** **Example:** `R4(config)# autonomic`	Configures R4 as an autonomic device.
Step 12	**exit** **Example:** `Device(config)# exit`	Exits global configuration mode and enters privileged EXEC mode.

What to Do Next

Enable *auto* mode on Auto-IP ring ports

Enabling Auto Mode on Auto-IP Ring Ports

Before You Begin

Identify the ports that will be part of the Auto-IP ring. Remember that you must enable Auto mode on all the ports in an Auto-IP ring.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp run**
4. **interface** *type number*
5. **auto-ip-ring** *ring-id* **ipv4-auto**
6. **exit**
7. Repeat steps to configure auto mode on each Auto-IP ring port.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**lldp run** **Example:** `Device(config)# lldp run`	Enables Link Layer Discovery Protocol (LLDP) for the device.
Step 4	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/0`	Specifies an interface type and number, and enters interface configuration mode.
Step 5	**auto-ip-ring** *ring-id* **ipv4-auto** **Example:** `Device(config-if)# auto-ip-ring 1 ipv4-auto`	Configures auto mode on the Auto-IP ring port.
Step 6	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 7	Repeat steps to configure auto mode on each Auto-IP ring port.	---

What to Do Next

Configure an Auto-IP server and reserve a pool of IP addresses for the Auto-IP ring ports.

Configuring an Auto-IP Server and Reserving a Pool of IP Addresses on the Server

Before You Begin

Ensure that all ports of the ring are identified and auto mode is enabled on the ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **auto-ip-ring server**
4. **ipv4-address-pool** *auto-ipv4-address number-of-subnets*
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**auto-ip-ring server** **Example:** `Device(config)# auto-ip-ring server`	Configures the device as an Auto-IP server and enters Auto-IP server configuration mode.
Step 4	**ipv4-address-pool** *auto-ipv4-address number-of-subnets* **Example:** `Device(config-auto-ip-server)# ipv4-address-pool 10.1.1.10 6`	Reserves a pool of IP addresses on the Auto-IP server. The number of subnets should, at a minimum, be the total number of owner ports or devices in the ring. The odd-numbered IP addresses are assigned to the owner ports, and each non owner port fetches its IP address from the owner port through LLDP
Step 5	**exit** **Example:** `Device(config-auto-ip-server)# exit`	Exits Auto-IP server configuration mode and enters global configuration mode.

What to Do Next

Configure a seed port to start the Auto-IP negotiation process.

Configuring a Seed Port

Before You Begin

Ensure all the Auto-IP ports are in auto mode, and a pool of IP addresses is reserved for the Auto-IP ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **auto-ip-ring** *ring-id* **ipv4-seed**
5. **exit**
6. **end**
7. **show auto-ip-ring** [*ring-id*][**detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/1`	Specifies an interface type and number, and enters interface configuration mode.
Step 4	**auto-ip-ring** *ring-id* **ipv4-seed** **Example:** `Device(config-if)# auto-ip-ring 1 ipv4-seed`	Designates the port as the seed port and initiates the Auto-IP negotiation process.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show auto-ip-ring** [*ring-id*][**detail**] **Example:** `Device# show auto-ip-ring 4 detail`	Displays auto-IP information.

What to Do Next

Verify if the IP addresses have been configured.

Verifying and Troubleshooting Zero Touch Auto-IP

Perform this task to verify Zero touch Auto-IP functions.

Note The commands are not in any specific order.

SUMMARY STEPS

1. **enable**
2. **show auto-ip-ring** [*ring-id*][**detail**]
3. **show autonomic service**
4. **show autonomic device**
5. **show autonomic neighbors**
6. **debug auto-ip-ring** {*ring-id* {**errors** | **events**} |**errors** | **events**}

DETAILED STEPS

Step 1 **enable**

Enables privileged EXEC mode.

Example:

```
Device> enable
```

Step 2 **show auto-ip-ring** [*ring-id*][**detail**]

This command displays Auto-IP ring information for a specific device or Auto-IP ring. The sample output given below displays two ports representing a ring, their IP addresses, and the connected ports and IP addresses (neighboring port information is denoted by Rx).

Example:

```
Device# show auto-ip-ring 1

Auto-IP ring 1

 Auto-IP Address          : 10.1.1.11

 Ring Port0               : Ethernet0/1
 My Current-IP            : 10.1.1.11
 My Priority              : 2

 Rx Auto-IP Address       : 10.1.1.13
 Rx Current-IP            : 10.1.1.12
 Rx-Priority              : 0

 Ring Port1               : Ethernet0/0
 My Current-IP            : 10.1.1.10
 My Priority              : 0

 Rx Auto-IP Address       : 10.1.1.17
 Rx Current-IP            : 10.1.1.17
 Rx-Priority              : 2
```

Step 3 **show autonomic service**

The following is sample output from this command, and it displays autonomic services configured on a device connected to an autonomic network.

Example:

```
Device# show autonomic service

Service                        IP-Addr
Autonomic registrar            FD53:EE55:A541:0:AABB:CC00:100:1
ANR type                       IOS CA
Auto IP Server                 FD53:EE55:A541:0:AABB:CC00:100:1
```

Step 4 **show autonomic device**

The following is sample output from this command, and it displays autonomic network configuration credentials for a device that is connected to the autonomic network. Details like unique identifier (UDI), device identifier (Device ID), associated domain (Domain ID), and so on, are displayed.

Example:

```
Device# show autonomic device

        UDI                             PID:Unix SN:655773698
        Device ID                       aabb.cc00.0100-2
        Domain ID                       auto-networking.com
        Domain Certificate              (sub:) ou=abcd.com+serialNumber=PID:Unix
SN:655773698,cn=aabb.cc00.0100-2
        Certificate Serial Number       03
        Device Address                  FD53:EE55:A541:0:AABB:CC00:100:2
        Domain Cert is Valid
```

Step 5 **show autonomic neighbors**

The following is sample output from this command, and it displays autonomic configuration details of connected, neighbor devices. Details such as unique identifier (UDI), device identifier (Device ID), and associated domain (Domain ID), are displayed.

Example:

```
Device# show autonomic neighbors

        UDI                                          Device-ID             Domain     Interface
--------------------------------------------------------------------------------
PID:Unix SN:655773697                         aabb.cc00.0100-1      abcd.com
Ethernet0/0
PID:Unix SN:655773699                         aabb.cc00.0100-4      abcd.com   Ethernet0/1
```

Step 6 **debug auto-ip-ring** {*ring-id* {**errors** | **events**} |**errors** | **events**}

The following is sample output from this command, and it displays debug errors and events for the specified Auto-IP ring.

Note A conflict is detected in the sample debug output below because the priority in the Auto-IP Type Length Value (TLV) that is sent from the interface and the priority that is received from the neighbor interface are the same.

Example:

```
Device# debug auto-ip-ring 2 errors

Auto IP Ring errors debugging is on for the ring id : 2
*Jul 26 11:30:40.541: (Ethernet0/0) priority (value:1) conflict detected,  need admin intervention
```

Configuration Examples for Zero Touch Auto-IP

Example: Associating an Auto-IP Server with an Autonomic Network

Auto-IP server (R1) is associated with the autonomic network. The other devices in the network (R2, R3, and R4) are enabled with the autonomic status.

```
R1(config)# autonomic registrar
R1(config-registrar)# domain-id auto-addressing.com
R1(config-registrar)# no shutdown
R1(config-registrar)# CA local
R1(config-registrar)# exit
R1(config)# autonomic


R2(config)# autonomic
R3(config)# autonomic
R4(config)# autonomic
```

Example: Enabling Auto Mode on Auto-IP Ring Ports

```
Device> enable
Device# configure terminal
Device(config)# lldp run
Device(config)# interface ethernet 0/0
Device(config-if)# auto-ip-ring 1 ipv4-auto
Device(config-if)# exit

Repeat the preceding steps to configure the auto mode on each Auto-IP ring port
```

Example: Configuring an Auto-IP Server and Reserving a Pool of IP Addresses on the Server

```
Device> enable
Device# configure terminal
Device(config)# auto-ip-ring server
Device(config-auto-ip-server)# ipv4-address-pool 10.1.1.10 6
Device(config-auto-ip-server)# exit
```

Example: Configuring a Seed Port

```
Device> enable
Device# configure terminal
Device(config)# interface e0/0
Device(config-if)# auto-ip-ring 1 ipv4-seed
Device(config-if)# exit
```

Additional References for Zero Touch Auto-IP

Related Documents

Related Topic	Document Title
Auto-IP	IP Addressing: IPv4 Addressing Configuration Guide
Configuring IPv4 Addresses	IP Addressing: IPv4 Addressing Configuration Guide
Using Link Layer Discovery Protocol in Multivendor Networks	Carrier Ethernet Configuration Guide
IPv4 Addressing commands	Cisco IOS IP Addressing Services Command Reference
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Auto-IP

Table 14: Feature Information for Auto-IP

Feature Name	Releases	Feature Information
Zero Touch Auto-IP		The Zero Touch Auto-IP feature enables automatic allocation and configuration of IP addresses for nodes in an Auto-IP ring. The IP addresses are allocated from a pool of IP addresses. The following commands were introduced or modified: **auto-ip-ring ipv4-auto**, **auto-ip-ring ipv4-seed**, **auto-ip-ring server**, **ipv4-address-pool**.