

Network Configuration Example

Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses

Release
12.3



Published: 2012-11-14

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Network Configuration Example Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses

Release 12.3

Copyright © 2012, Juniper Networks, Inc.

All rights reserved.

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

Introduction	1
Advantages of Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses	1
Understanding Dual Stacking	1
Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses	3

Introduction

This document explains how to use dual stacking to simultaneously use IPv4 addresses and IPv6 addresses. An example video to show how to configure both addresses is also included.

Advantages of Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses

If you are a service provider, you always need to add more addresses to your network. Because the IPv4 addresses have all been allocated, you need a way to add IPv6 addresses while maintaining your investment in the IPv4 infrastructure.

Dual stacking offers a way to configure both IPv6 and IPv4 simultaneously. This allows you to continue to add more subscribers even after you run out of IPv4 addresses. If you are a service provider, dual stacking can help you reduce your operational expenses and increase your capital.

Dual stacking helps to protect the investment in your existing infrastructure, the devices that support IPv4. It also allows you to add new subscribers that are using IPv6-enabled devices.

Dual stacking allows you to have more subscribers on your network, and make use of your IPv4 systems without having a lot of replacement costs. Going forward, you can purchase devices that only support IPv6.

If you are a service provider, you can expand your service offerings to customers using IPv6 and still accommodate dual addressing. You will be able to support more subscribers on your network with less work.

Devices that support dual stacking enable your business to grow because you can offer expanded services that allow you to generate more revenue.

Related Documentation

- Understanding IPv6 Dual-Stack Lite
- Learn About: IPv6
- [Understanding Dual Stacking on page 1](#)
- [Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses on page 3](#)

Understanding Dual Stacking

Service providers and enterprises are faced with growing their networks using IPv6, while continuing to serve IPv4 customers.

Increasingly, the public side of network address translation (NAT) devices is IPv6 rather than IPv4. Service providers cannot continue giving customers globally routable IPv4 addresses, they cannot get new globally routable IPv4 addresses for expanding their own networks, and yet they must continue to serve both IPv4 customers and new customers, all of whom are primarily trying to reach IPv4 destinations.

IPv4 and IPv6 must coexist for some number of years, and their coexistence must be transparent to end users. If an IPv4-to-IPv6 transition is successful, end users should not even notice it.

A dual-stack device is a device with network interfaces that can originate and understand both IPv4 and IPv6 packets.

Other strategies, such as manually or dynamically configured tunnels and translation devices exist, but dual stacking is often the preferable solution in many scenarios. The dual-stacked device can interoperate equally with IPv4 devices, IPv6 devices, and other dual-stacked devices. When both devices are dual stacked, the two devices agree on which IP version to use.

The transition is driven by DNS. If a dual-stacked device queries the name of a destination and DNS gives it an IPv4 address (a DNS A Record), it sends IPv4 packets. If DNS responds with an IPv6 address (a DNS AAAA Record), it sends IPv6 packets.

Keep in mind that if you are going to dual stack all of your network devices, the interfaces need both an IPv6 and an IPv4 address. This raises the issue that the Internet has run out of IPv4 addresses, which is the main reason IPv6 is needed in the first place. If you do not have an abundant supply of IPv4 addresses to apply to your devices, you can still use dual stacking, but you will need to conserve your supply of IPv4 addresses by using network address translation (NAT). Building dual-stacked networks with a mix of global IPv6 addresses and NAT-ed IPv4 addresses is quite feasible. Some specific solutions include carrier-grade NAT (CGN), NAT44(4), NAT64, NAT464, and dual-stack lite.

[Table 1 on page 2](#) describes at a high level how to pick a network addressing technique. In reality, a complete solution might include a set of techniques to satisfy multiple service needs. It is important to understand the backbone technology being used on the network and also to know if the provider has control over the access customer premises equipment (CPE).

Table 1: Choosing the Right Solution to Address Next-Generation Addressing Requirements

CPE Network	Access Network	Destinations	Solution
IPv4	IPv4	IPv4 Internet	NAT44(4)
IPv4/IPv6	IPv6	IPv4 Internet	DS-Lite with NAT44
IPv4/IPv6	IPv4	IPv6 Internet	6rd (6to4)
IPv4	IPv6	IPv4 Internet	NAT64

Related Documentation

- Understanding IPv6 Dual-Stack Lite
- Learn About: IPv6
- [Advantages of Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses on page 1](#)

- [Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses on page 3](#)

Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses

This example shows how to configure IPv4 and IPv6 dual stacking in IS-IS.

- [Requirements on page 3](#)
- [Overview on page 3](#)
- [Configuration on page 4](#)
- [Verification on page 6](#)

Requirements

No special configuration beyond device initialization is required before configuring this example.

Overview



VIDEO: [IS-IS Dual Stacking](#)

You can use IPv4 and IPv6 dual stacking to begin your migration from IPv4 to IPv6 by implementing IPv6 alongside IPv4 in your existing networks. This allows you to implement IPv6 so that you can provide the same services over IPv6—for example, video, voice, high-quality data—that you currently provide in your IPv4 networks. You can then perform incremental upgrades to IPv6 and avoid service disruptions while migrating from IPv4 to IPv6.

Unlike RIP and OSPF, IS-IS does not require a distinct protocol or a new version to support IPv6. Because IS-IS uses ISO addresses, the configuration for IPv6 and IPv4 is identical in the Junos OS implementation of IS-IS. For IS-IS to carry IPv6 routes, you only need to add IPv6 addresses to IS-IS enabled interfaces or include other IPv6 routes in your IS-IS export policy.

The only explicit configuration needed in IS-IS with regard to IPv6 is if you want to disable it. Alternatively, you can disable IPv4 routing and use IS-IS with IPv6 only. An example of each is provided here:

Disable IPv6 routing in IS-IS:

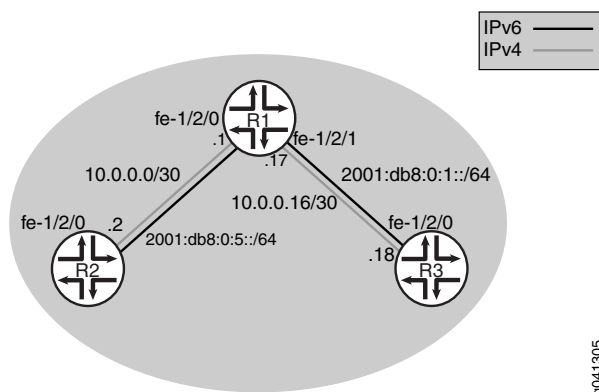
```
[edit protocols isis]  
user@host# set no-ipv6-routing
```

Use IS-IS exclusively for IPv6 routing:

```
[edit protocols isis]  
user@host# set no-ipv4-routing
```

[Figure 1 on page 4](#) shows the topology used in this example.

Figure 1: IS-IS IPv4 and IPv6 Dual Stacking Topology



“CLI Quick Configuration” on page 4 shows the configuration for all of the devices in Figure 1 on page 4. The section “Step-by-Step Procedure” on page 5 describes the steps on Device R1.

Configuration

CLI Quick Configuration To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

Device R1

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.1/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces fe-1/2/1 unit 0 family inet address 10.0.0.17/30
set interfaces fe-1/2/1 unit 0 family iso
set interfaces fe-1/2/1 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::1/128
set protocols isis interface fe-1/2/0.0
set protocols isis interface fe-1/2/1.0
set protocols isis interface lo0.0

```

Device R2

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.2/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0002.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::2/128
set protocols isis interface fe-1/2/0.0
set protocols isis interface lo0.0

```

Device R3

```

set interfaces fe-1/2/0 unit 0 family inet address 10.0.0.18/30
set interfaces fe-1/2/0 unit 0 family iso
set interfaces fe-1/2/0 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set interfaces lo0 unit 0 family iso address 49.0002.0192.0168.0003.00
set interfaces lo0 unit 0 family inet6 address 2001:db8::3/128

```

```
set protocols isis interface fe-1/2/0.0
set protocols isis interface lo0.0
```

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see Using the CLI Editor in Configuration Mode in the Junos OS CLI User Guide.

To configure IS-IS dual stacking:

1. Configure the interfaces, including both IPv4 and IPv6 addresses on each interface.

Optionally, include the **eui-64** statement to automatically generate the host number portion of interface addresses.

```
[edit interfaces]
user@R1# set fe-1/2/0 unit 0 family inet address 10.0.0.1/30
user@R1# set fe-1/2/0 unit 0 family iso
user@R1# set fe-1/2/0 unit 0 family inet6 address 2001:db8:0:5::/64 eui-64
```

```
user@R1# set fe-1/2/1 unit 0 family inet address 10.0.0.17/30
user@R1# set fe-1/2/1 unit 0 family iso
user@R1# set fe-1/2/1 unit 0 family inet6 address 2001:db8:0:1::/64 eui-64
```

```
user@R1# set lo0 unit 0 family inet address 192.168.0.1/32
user@R1# set lo0 unit 0 family iso address 49.0002.0192.0168.0001.00
user@R1# set lo0 unit 0 family inet6 address 2001:db8::1/128
```

2. Enable IS-IS on the interfaces.

```
[edit protocols isis]
user@R1# set interface fe-1/2/0.0
user@R1# set interface fe-1/2/1.0
user@R1# set interface lo0.0
```

Results From configuration mode, confirm your configuration by entering the **show interfaces** and **show protocols** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
fe-1/2/0 {
  unit 0 {
    family inet {
      address 10.0.0.1/30;
    }
    family iso;
    family inet6 {
      address 2001:db8:0:5::/64 {
        eui-64;
      }
    }
  }
}
fe-1/2/1 {
  unit 0 {
    family inet {
```

```
        address 10.0.0.17/30;
    }
    family iso;
    family inet6 {
        address 2001:db8:0:1::/64 {
            eui-64;
        }
    }
}
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.0.1/32;
        }
        family iso {
            address 49.0002.0192.0168.0001.00;
        }
        family inet6 {
            address 2001:db8::1/128;
        }
    }
}
```

user@R1# show protocols

```
isis {
    interface fe-1/2/0.0;
    interface fe-1/2/1.0;
    interface lo0.0;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- [Checking the Neighbor Adjacencies on page 6](#)
- [Pinging the IPv6 Interfaces on page 7](#)
- [Checking the IPv6 Routing Table on page 8](#)

Checking the Neighbor Adjacencies

Purpose Determine what topologies are supported on neighboring IS-IS devices.

Action From operational mode, enter the **show isis adjacency detail** command.

```
user@R1> show isis adjacency detail
```

```
R2
```

```
Interface: fe-1/2/0.0, Level: 3, State: Up, Expires in 24 secs
Priority: 0, Up/Down transitions: 1, Last transition: 18:34:08 ago
Circuit type: 3, Speaks: IP, IPv6
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.0.0.2
```

IPv6 addresses: fe80::2a0:a514:0:24c

R3

Interface: fe-1/2/1.0, Level: 3, State: Up, Expires in 21 secs
Priority: 0, Up/Down transitions: 1, Last transition: 18:33:41 ago
Circuit type: 3, **Speaks: IP, IPv6**
Topologies: Unicast
Restart capable: Yes, Adjacency advertisement: Advertise
IP addresses: 10.0.0.18
IPv6 addresses: fe80::2a0:a514:0:124c

Meaning As expected, the output shows that the two neighbors support both IPv4 and IPv6. The IPv4 address and the IPv6 link-local address are also shown.

[Pinging the IPv6 Interfaces](#)

Purpose Make sure that you can ping the remote IPv6 interfaces.

Action From operational mode, enter the **ping** command to ping from Device R2 to Device R3.

1. Determine the IPv6 address assigned to Device R3.

If you use EUI-64 addressing as shown in the example, the host portion of the IPv6 addresses is assigned automatically. To determine what addresses are assigned, use the **show interfaces terse** command on Device R3.

```
user@R3> show interfaces terse
Interface           Admin Link Proto  Local                    Remote
fe-1/2/0
fe-1/2/0.0          up    up    inet   10.0.0.18/30
                   iso
                   inet6   2001:db8:0:1:2a0:a514:0:124c/64
                   fe80::2a0:a514:0:124c/64

lo0
lo0.0               up    up    inet   192.168.0.3             --> 0/0
                   iso
                   inet6   2001:db8::3
                   fe80::2a0:a50f:fc56:14c
```

The IPv6 addresses that should be pingable are 2001:db8:0:1:2a0:a514:0:124c and 2001:db8::3.

2. From Device R2, ping the Device R3 fe-1/2/0.0 IPv6 interface address and the lo0.0 IPv6 interface address.

```
user@R2> ping 2001:db8:0:1:2a0:a514:0:124c
PING6(56=40+8+8 bytes) 2001:db8:0:5:2a0:a514:0:24c -->
2001:db8:0:1:2a0:a514:0:124c
16 bytes from 2001:db8:0:1:2a0:a514:0:124c, icmp_seq=0 hlim=63 time=2.373 ms
16 bytes from 2001:db8:0:1:2a0:a514:0:124c, icmp_seq=1 hlim=63 time=1.600 ms
16 bytes from 2001:db8:0:1:2a0:a514:0:124c, icmp_seq=2 hlim=63 time=2.228 ms

user@R2> ping 2001:db8::3
PING6(56=40+8+8 bytes) 2001:db8:0:5:2a0:a514:0:24c --> 2001:db8::3
16 bytes from 2001:db8::3, icmp_seq=0 hlim=63 time=1.797 ms
16 bytes from 2001:db8::3, icmp_seq=1 hlim=63 time=1.430 ms
16 bytes from 2001:db8::3, icmp_seq=2 hlim=63 time=2.525 ms
```

Meaning This test confirms that IS-IS has learned the IPv6 routes.

Checking the IPv6 Routing Table

Purpose Verify that the expected routes are in the IPv6 routing table.

Action

```
user@R1> show route table inet6.0
inet6.0: 11 destinations, 12 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8::1/128    *[Direct/0] 18:52:52
                  > via lo0.0
2001:db8::2/128    *[IS-IS/15] 01:59:52, metric 10
                  > to fe80::2a0:a514:0:24c via fe-1/2/0.0
2001:db8::3/128    *[IS-IS/15] 01:59:52, metric 10
                  > to fe80::2a0:a514:0:124c via fe-1/2/1.0
2001:db8:0:1::/64  *[Direct/0] 18:52:15
                  > via fe-1/2/1.0
2001:db8:0:1:2a0:a514:0:114c/128
                  *[Local/0] 18:52:48
                  Local via fe-1/2/1.0
2001:db8:0:5::/64  *[Direct/0] 18:52:49
                  > via fe-1/2/0.0
2001:db8:0:5:2a0:a514:0:14c/128
                  *[Local/0] 18:52:49
                  Local via fe-1/2/0.0
fe80::/64          *[Direct/0] 18:52:49
                  > via fe-1/2/0.0
                  [Direct/0] 18:52:15
                  > via fe-1/2/1.0
fe80::2a0:a50f:fc56:14c/128
                  *[Direct/0] 18:52:52
                  > via lo0.0
fe80::2a0:a514:0:14c/128
                  *[Local/0] 18:52:49
                  Local via fe-1/2/0.0
fe80::2a0:a514:0:114c/128
                  *[Local/0] 18:52:48
                  Local via fe-1/2/1.0
```

Meaning The output shows the IPv6 interface routes (direct and local) and the IPv6 routes learned through IS-IS.

Related Documentation

- [Example: Configuring IS-IS IPv4 and IPv6 Unicast Topologies](#)
- [Advantages of Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses on page 1](#)
- [Example: Configuring IS-IS Dual Stacking of IPv4 and IPv6 Unicast Addresses on page 3](#)