



Junos[®] OS

Multitopology Routing Feature Guide

Release

11.1



Published: 2011-01-19

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.

Junos® OS Multitopology Routing Feature Guide

Release 11.1

Copyright © 2011, Juniper Networks, Inc.

All rights reserved. Printed in USA.

Revision History

January 2011—R1 Junos OS 11.1

The information in this document is current as of the date listed in the revision history.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. The 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

READ THIS END USER LICENSE AGREEMENT ("AGREEMENT") BEFORE DOWNLOADING, INSTALLING, OR USING THE SOFTWARE. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE OR OTHERWISE EXPRESSING YOUR AGREEMENT TO THE TERMS CONTAINED HEREIN, YOU (AS CUSTOMER OR IF YOU ARE NOT THE CUSTOMER, AS A REPRESENTATIVE/AGENT AUTHORIZED TO BIND THE CUSTOMER) CONSENT TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT OR CANNOT AGREE TO THE TERMS CONTAINED HEREIN, THEN (A) DO NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND (B) YOU MAY CONTACT JUNIPER NETWORKS REGARDING LICENSE TERMS.

1. **The Parties.** The parties to this Agreement are (i) Juniper Networks, Inc. (if the Customer's principal office is located in the Americas) or Juniper Networks (Cayman) Limited (if the Customer's principal office is located outside the Americas) (such applicable entity being referred to herein as "Juniper"), and (ii) the person or organization that originally purchased from Juniper or an authorized Juniper reseller the applicable license(s) for use of the Software ("Customer") (collectively, the "Parties").

2. **The Software.** In this Agreement, "Software" means the program modules and features of the Juniper or Juniper-supplied software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller, or which was embedded by Juniper in equipment which Customer purchased from Juniper or an authorized Juniper reseller. "Software" also includes updates, upgrades and new releases of such software. "Embedded Software" means Software which Juniper has embedded in or loaded onto the Juniper equipment and any updates, upgrades, additions or replacements which are subsequently embedded in or loaded onto the equipment.

3. **License Grant.** Subject to payment of the applicable fees and the limitations and restrictions set forth herein, Juniper grants to Customer a non-exclusive and non-transferable license, without right to sublicense, to use the Software, in executable form only, subject to the following use restrictions:

- a. Customer shall use Embedded Software solely as embedded in, and for execution on, Juniper equipment originally purchased by Customer from Juniper or an authorized Juniper reseller.
- b. Customer shall use the Software on a single hardware chassis having a single processing unit, or as many chassis or processing units for which Customer has paid the applicable license fees; provided, however, with respect to the Steel-Belted Radius or Odyssey Access Client software only, Customer shall use such Software on a single computer containing a single physical random access memory space and containing any number of processors. Use of the Steel-Belted Radius or IMS AAA software on multiple computers or virtual machines (e.g., Solaris zones) requires multiple licenses, regardless of whether such computers or virtualizations are physically contained on a single chassis.
- c. Product purchase documents, paper or electronic user documentation, and/or the particular licenses purchased by Customer may specify limits to Customer's use of the Software. Such limits may restrict use to a maximum number of seats, registered endpoints, concurrent users, sessions, calls, connections, subscribers, clusters, nodes, realms, devices, links, ports or transactions, or require the purchase of separate licenses to use particular features, functionalities, services, applications, operations, or capabilities, or provide throughput, performance, configuration, bandwidth, interface, processing, temporal, or geographical limits. In addition, such limits may restrict the use of the Software to managing certain kinds of networks or require the Software to be used only in conjunction with other specific Software. Customer's use of the Software shall be subject to all such limitations and purchase of all applicable licenses.
- d. For any trial copy of the Software, Customer's right to use the Software expires 30 days after download, installation or use of the Software. Customer may operate the Software after the 30-day trial period only if Customer pays for a license to do so. Customer may not extend or create an additional trial period by re-installing the Software after the 30-day trial period.
- e. The Global Enterprise Edition of the Steel-Belted Radius software may be used by Customer only to manage access to Customer's enterprise network. Specifically, service provider customers are expressly prohibited from using the Global Enterprise Edition of the Steel-Belted Radius software to support any commercial network access services.

The foregoing license is not transferable or assignable by Customer. No license is granted herein to any user who did not originally purchase the applicable license(s) for the Software from Juniper or an authorized Juniper reseller.

4. **Use Prohibitions.** Notwithstanding the foregoing, the license provided herein does not permit the Customer to, and Customer agrees not to and shall not: (a) modify, unbundle, reverse engineer, or create derivative works based on the Software; (b) make unauthorized copies of the Software (except as necessary for backup purposes); (c) rent, sell, transfer, or grant any rights in and to any copy of the Software, in any form, to any third party; (d) remove any proprietary notices, labels, or marks on or in any copy of the Software or any product in which the Software is embedded; (e) distribute any copy of the Software to any third party, including as may be embedded in Juniper equipment sold in the secondhand market; (f) use any 'locked' or key-restricted feature, function, service, application, operation, or capability without first purchasing the applicable license(s) and obtaining a valid key from Juniper, even if such feature, function, service, application, operation, or capability is enabled without a key; (g) distribute any key for the Software provided by Juniper to any third party; (h) use the

Software in any manner that extends or is broader than the uses purchased by Customer from Juniper or an authorized Juniper reseller; (i) use Embedded Software on non-Juniper equipment; (j) use Embedded Software (or make it available for use) on Juniper equipment that the Customer did not originally purchase from Juniper or an authorized Juniper reseller; (k) disclose the results of testing or benchmarking of the Software to any third party without the prior written consent of Juniper; or (l) use the Software in any manner other than as expressly provided herein.

5. **Audit.** Customer shall maintain accurate records as necessary to verify compliance with this Agreement. Upon request by Juniper, Customer shall furnish such records to Juniper and certify its compliance with this Agreement.

6. **Confidentiality.** The Parties agree that aspects of the Software and associated documentation are the confidential property of Juniper. As such, Customer shall exercise all reasonable commercial efforts to maintain the Software and associated documentation in confidence, which at a minimum includes restricting access to the Software to Customer employees and contractors having a need to use the Software for Customer's internal business purposes.

7. **Ownership.** Juniper and Juniper's licensors, respectively, retain ownership of all right, title, and interest (including copyright) in and to the Software, associated documentation, and all copies of the Software. Nothing in this Agreement constitutes a transfer or conveyance of any right, title, or interest in the Software or associated documentation, or a sale of the Software, associated documentation, or copies of the Software.

8. **Warranty, Limitation of Liability, Disclaimer of Warranty.** The warranty applicable to the Software shall be as set forth in the warranty statement that accompanies the Software (the "Warranty Statement"). Nothing in this Agreement shall give rise to any obligation to support the Software. Support services may be purchased separately. Any such support shall be governed by a separate, written support services agreement. TO THE MAXIMUM EXTENT PERMITTED BY LAW, JUNIPER SHALL NOT BE LIABLE FOR ANY LOST PROFITS, LOSS OF DATA, OR COSTS OR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT, THE SOFTWARE, OR ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. IN NO EVENT SHALL JUNIPER BE LIABLE FOR DAMAGES ARISING FROM UNAUTHORIZED OR IMPROPER USE OF ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. EXCEPT AS EXPRESSLY PROVIDED IN THE WARRANTY STATEMENT TO THE EXTENT PERMITTED BY LAW, JUNIPER DISCLAIMS ANY AND ALL WARRANTIES IN AND TO THE SOFTWARE (WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE), INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT DOES JUNIPER WARRANT THAT THE SOFTWARE, OR ANY EQUIPMENT OR NETWORK RUNNING THE SOFTWARE, WILL OPERATE WITHOUT ERROR OR INTERRUPTION, OR WILL BE FREE OF VULNERABILITY TO INTRUSION OR ATTACK. In no event shall Juniper's or its suppliers' or licensors' liability to Customer, whether in contract, tort (including negligence), breach of warranty, or otherwise, exceed the price paid by Customer for the Software that gave rise to the claim, or if the Software is embedded in another Juniper product, the price paid by Customer for such other product. Customer acknowledges and agrees that Juniper has set its prices and entered into this Agreement in reliance upon the disclaimers of warranty and the limitations of liability set forth herein, that the same reflect an allocation of risk between the Parties (including the risk that a contract remedy may fail of its essential purpose and cause consequential loss), and that the same form an essential basis of the bargain between the Parties.

9. **Termination.** Any breach of this Agreement or failure by Customer to pay any applicable fees due shall result in automatic termination of the license granted herein. Upon such termination, Customer shall destroy or return to Juniper all copies of the Software and related documentation in Customer's possession or control.

10. **Taxes.** All license fees payable under this agreement are exclusive of tax. Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software. If applicable, valid exemption documentation for each taxing jurisdiction shall be provided to Juniper prior to invoicing, and Customer shall promptly notify Juniper if their exemption is revoked or modified. All payments made by Customer shall be net of any applicable withholding tax. Customer will provide reasonable assistance to Juniper in connection with such withholding taxes by promptly: providing Juniper with valid tax receipts and other required documentation showing Customer's payment of any withholding taxes; completing appropriate applications that would reduce the amount of withholding tax to be paid; and notifying and assisting Juniper in any audit or tax proceeding related to transactions hereunder. Customer shall comply with all applicable tax laws and regulations, and Customer will promptly pay or reimburse Juniper for all costs and damages related to any liability incurred by Juniper as a result of Customer's non-compliance or delay with its responsibilities herein. Customer's obligations under this Section shall survive termination or expiration of this Agreement.

11. **Export.** Customer agrees to comply with all applicable export laws and restrictions and regulations of any United States and any applicable foreign agency or authority, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals. Customer shall be liable for any such violations. The version of the Software supplied to Customer may contain encryption or other capabilities restricting Customer's ability to export the Software without an export license.

12. **Commercial Computer Software.** The Software is "commercial computer software" and is provided with restricted rights. Use, duplication, or disclosure by the United States government is subject to restrictions set forth in this Agreement and as provided in DFARS 227.7201 through 227.7202-4, FAR 12.212, FAR 27.405(b)(2), FAR 52.227-19, or FAR 52.227-14 (ALT III) as applicable.

13. **Interface Information.** To the extent required by applicable law, and at Customer's written request, Juniper shall provide Customer with the interface information needed to achieve interoperability between the Software and another independently created program, on payment of applicable fee, if any. Customer shall observe strict obligations of confidentiality with respect to such information and shall use such information in compliance with any applicable terms and conditions upon which Juniper makes such information available.

14. **Third Party Software.** Any licensor of Juniper whose software is embedded in the Software and any supplier of Juniper whose products or technology are embedded in (or services are accessed by) the Software shall be a third party beneficiary with respect to this Agreement, and such licensor or vendor shall have the right to enforce this Agreement in its own name as if it were Juniper. In addition, certain third party software may be provided with the Software and is subject to the accompanying license(s), if any, of its respective owner(s). To the extent portions of the Software are distributed under and subject to open source licenses obligating Juniper to make the source code for such portions publicly available (such as the GNU General Public License ("GPL") or the GNU Library General Public License ("LGPL")), Juniper will make such source code portions (including Juniper modifications, as appropriate) available upon request for a period of up to three years from the date of distribution. Such request can be made in writing to Juniper Networks, Inc., 1194 N. Mathilda Ave., Sunnyvale, CA 94089, ATTN: General Counsel. You may obtain a copy of the GPL at <http://www.gnu.org/licenses/gpl.html>, and a copy of the LGPL at <http://www.gnu.org/licenses/lgpl.html>.

15. **Miscellaneous.** This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présentes confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattache, soient rédigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).

Table of Contents

Part 1	Multitopology Routing	
Chapter 1	Multitopology Routing Concept and Reference Materials	3
	Multitopology Routing Overview	3
	Multitopology Routing System Requirements	5
	Multitopology Routing Terms and Acronyms	5
Chapter 2	Multitopology Routing Configuration	7
	Configuring Topologies	7
	Configuring Filter-Based Forwarding	7
	Configuring BGP for Multitopology Routing	8
	Option: Configuring OSPF for Multitopology Routing	8
	Option: Configuring Static Routes for Multitopology Routing	9
	Option: Configuring Route Resolution Policy	9
Chapter 3	Multitopology Routing Configuration Examples	11
	Example: Multitopology Routing Configuration	11
	Verifying Your Work	15
	For More Information	15
Part 2	Index	
	Index	19

List of Figures

Part 1	Multitopology Routing	
Chapter 1	Multitopology Routing Concept and Reference Materials	3
	Figure 1: MTR-OSPF Area Boundary	4
	Figure 2: BGP Route Resolution	4
	Figure 3: Route Resolution for MTR	5
Chapter 3	Multitopology Routing Configuration Examples	11
	Figure 4: Route Resolution in Multitopology Routing	12

PART 1

Multitopology Routing

- Multitopology Routing Concept and Reference Materials on page 3
- Multitopology Routing Configuration on page 7
- Multitopology Routing Configuration Examples on page 11

CHAPTER 1

Multitopology Routing Concept and Reference Materials

This chapter covers these topics:

- Multitopology Routing Overview on page 3
- Multitopology Routing System Requirements on page 5
- Multitopology Routing Terms and Acronyms on page 5

Multitopology Routing Overview

Multitopology Routing (MTR) enables you to configure class-based forwarding for different types of traffic, such as voice, video, and data. Each type of traffic is defined by a topology that is used to create a new routing table for that topology. MTR provides the ability to generate forwarding tables based on the resolved entries in the routing tables for the custom topologies you create. In this way, packets of different classes can be routed independently from one another.

To run MTR, you must configure IP routing. MTR supports OSPFv2, static routes, and BGP. You must configure an interior gateway protocol (IGP), such as OSPFv2 or static routing. Configure BGP to add routes learned through BGP to the appropriate custom topologies. MTR also supports filter-based forwarding, which enables you to match traffic on the ingress interface with a specific type of forwarding class and then forward that traffic to the specified topology.

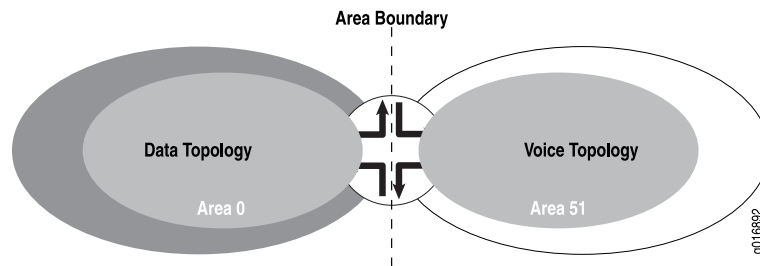
OSPF in MTR

OSPF in MTR uses a single instance of OSPF to carry connectivity and IP reachability information for different topologies. That information is used to calculate shortest-path first (SPF) trees and routing tables. OSPF for MTR supports protocol extensions that include metrics that correspond to different topologies for link and prefix reachability information. The type-of-service (TOS) metric field is used to advertise the topology-specific metric for links and prefixes belonging to that topology. The TOS field is redefined as MT-ID in the payload of router, summary, and Type 5 and Type 7 AS-external link-state advertisements (LSAs).

Under MTR, each OSPF interface continues to belong to a single area. Therefore, by default, all topologies share the same area boundaries. As a result, attributes of an area,

such as stubbiness, are independent of the topology. By default, all topologies configured for OSPF are enabled on all interfaces. However, you can disable one or more configured topologies on an interface. You can thus allocate an interface for a specific topology. In Figure 1 on page 4, Area 51 includes an interface that is uniquely allocated to voice traffic, and Area 0 includes an interface that is uniquely allocated to data traffic. Each topology thus corresponds to a different OSPF area that shares a boundary.

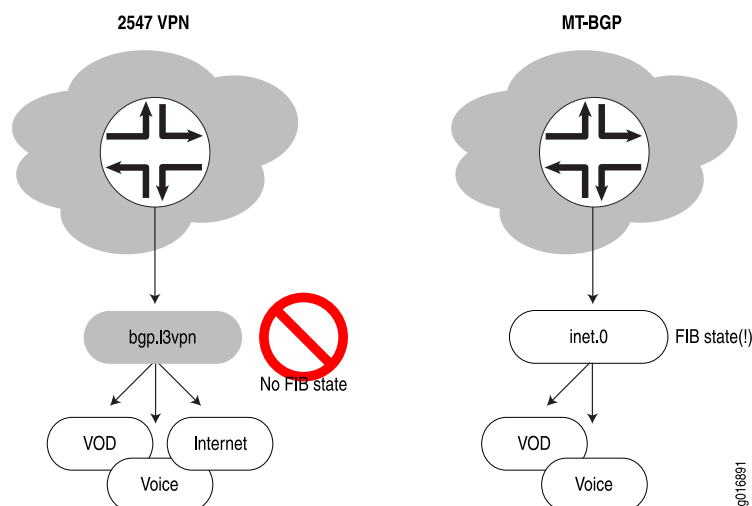
Figure 1: MTR-OSPF Area Boundary



BGP in MTR

BGP in MTR provides the ability to resolve BGP routes against configured topologies. An inbound policy is used to select routes for inclusion in the appropriate routing tables for the topologies. The default behavior for virtual private networks (VPNs) that use Multiprotocol Label Switching (MPLS) for forwarding packets over the backbone and BGP for distributing routes over the backbone is to place BGP route updates in the **bgp.l3vpn** routing table. Figure 2 on page 4 shows a BGP peer operating in an environment that conforms with the requirements in RFC 2547, *BGP/MPLS VPNs*. The figure shows how a BGP peer configured for MTR performs secondary route resolution.

Figure 2: BGP Route Resolution

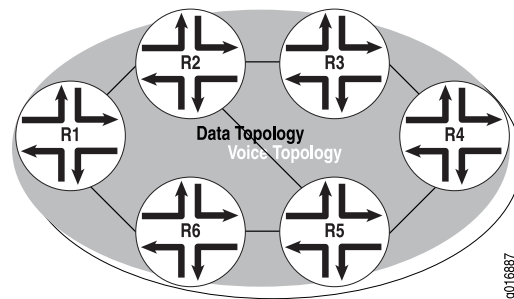


The BGP peer in a standard VPN topology places prefixes for routes it learns in the **bgp.l3vpn** routing table, which does not result in automatic updates to the forwarding table. Under BGP in MTR, when BGP receives a route from a peer it attempts to resolve that route against a route in the **inet.0** routing table. If the route is resolved, it is placed in that table, which generates a forwarding state. If you have configured a community

target identifier that matches the import policy for the topology, routing and forwarding states are added to the tables for the topology.

Because MTR provides support for BGP to perform secondary route resolution, as Figure 3 on page 5 shows, MTR is able to create two distinct network paths for each type of traffic. Each router advertises BGP routes that need to be resolved against the interior gateway protocol (IGP) routes for each topology. Based on the IGP metrics configured for each topology, for all routes that originate from Router 4 (R4), the upper path between R1 and R4, which traverses R2 and R3, is selected for voice traffic, whereas the lower path between R1 and R4, which traverses R5 and R6, is selected for data traffic.

Figure 3: Route Resolution for MTR



Multitopology Routing System Requirements

To implement MTR, your system must meet these minimum requirements:

- Junos OS Release 9.0 or later
- Two Juniper Networks M Series, MX Series, or T Series routers

Multitopology Routing Terms and Acronyms

T

topology	A subset of links in a network for which a separate set of routes is calculated. Those routes are installed in a routing table created specifically for a configured topology used to make routing and forwarding decisions.
-----------------	--

CHAPTER 2

Multitopology Routing Configuration

To implement MTR, you must configure the following:

- Configuring Topologies on page 7
- Configuring Filter-Based Forwarding on page 7
- Configuring BGP for Multitopology Routing on page 8
- Option: Configuring OSPF for Multitopology Routing on page 8
- Option: Configuring Static Routes for Multitopology Routing on page 9
- Option: Configuring Route Resolution Policy on page 9

Configuring Topologies

You must configure one or more topologies. For each topology, you specify a string value that defines the type of traffic as well as an interface family. You can also enable a topology for IPv4 multicast traffic by including the **ipv4-multicast** statement. To configure a topology, include the **topologies** statement at the **[edit routing-options]** hierarchy level:

```
[edit routing-options]
topologies {
  family inet { # inet6 is also supported, but you must use static routes as the IGP.
    topology voice; # This action creates a routing table called :voice.inet.0
      # for all routes destined for the voice topology. A default topology is also
      # automatically created. Default topology routes are added to the inet.0
      # routing table.
    }
  }
}
```

Configuring Filter-Based Forwarding

Configure a firewall filter that forces a lookup against the different routing tables. Any routes that match the specified forwarding class specified and then match a specified topology are installed in the routing table for that topology. To configure a firewall filter for MTR that performs filter-based forwarding, include the following configuration at the **[edit firewall]** hierarchy level:

```
[edit firewall]
family inet { # inet6 is also supported.
  filter topology-selection {
    term ef {
```

```
from {
    forwarding-class expedited-forwarding; # The following class types are also
    # supported: assured-forwarding, best-effort, and network-control.
}
then {
    topology voice; # Specify the name of a configured topology.
    accept;
}
}
}
```

You must apply the filter to an ingress interface. Include the following statements at the **[edit interfaces]** hierarchy level to apply the filter to an interface:

```
[edit interfaces]
fe-2/2/1 {
    unit 0 {
        family inet {
            filter {
                input topology-selection; # Specify the name of the filter configured under
                # the [edit firewall] hierarchy.
            }
        }
    }
}
```

Configuring BGP for Multitopology Routing

Configure BGP to add routes learned through BGP into a configured topology and the default topology, which is created automatically. Routes for the default topology are installed in the **inet.0** routing table. To configure BGP for MTR, include the following statements:

```
[edit protocols bgp]
group internal {
    type internal;
    family inet {
        unicast {
            topology voice;
            community target :1:1; # Any route that has :1:1 as its target destination is
            # installed in the routing table for the voice topology. All received routes
            # are also automatically installed in the default topology.
        }
    }
}
```

Option: Configuring OSPF for Multitopology Routing

To implement MTR, you must configure an interior gateway protocol (IGP) to route local network traffic. MTR supports both OSPFv2 and static routes. Only static routes support IPv6 addresses because MTR does not support OSPFv3. Configure OSPF to add routes

from the default topology to the routing table for the specified topology. To enable OSPF for MTR, include the **topology** statement at the **[edit protocols ospf]** hierarchy level:

```
[edit ospf protocols]
topology voice { # Specify a topology name configured under the [edit routing-options]
  # hierarchy level.
  topology-id 127; # Specify a topology identifier from 32 through 127.
}
```

Optionally, you can configure a specific metric for a topology for any interface on which OSPF has been enabled. Any topology-specific metric that you configure applies to routes advertised from that interface that belong only to that topology. To configure a topology-specific metric for an OSPF interface, include the following statements at the **[edit protocols ospf]** hierarchy level:

```
[edit protocols ospf]
area 0.0.0.0 {
  interface fe-2/2/1 {
    metric 10; # You can specify a metric for the interface that overrides the default
    # value of 1.
  }
  topology voice {
    metric 15; # Specify a topology-specific metric from 1 through 65,535.
  }
}
}
```

Option: Configuring Static Routes for Multitopology Routing

To configure a static route for MTR, you must specify the name of the routing table for the topology. Static routes for MTR support IPv4 and IPv6 addresses. To configure a static route for MTR, include the **rib** statement at the **[edit routing-options]** hierarchy level:

```
[edit routing-options]
rib :voice.inet6.0 { # Specify the routing-table name for the voice topology configured
  # at the [edit routing-options topologies] hierarchy level.
  route 200::a4:0/126 next-hop 200::c0a8:1df;
}
```

Option: Configuring Route Resolution Policy

You can optionally configure a route resolution policy so that a routing table accepts routes from specific routing tables. For MTR, you might want to configure a policy for the voice topology for IPv4 unicast traffic, for example, **:voice.inet.0** to resolve routes through the **inet.3** routing table. You might also want to override the default policy and not have the **inet.0** routing table use the **inet.3** routing table for route resolution. To configure a route resolution policy, include the **resolution** statement at the **[edit routing-options]** hierarchy level:

```
[edit routing-options]
resolution {
  rib inet.0 { # Specify the name of the routing table you want to modify.
    resolution-ribs inet.0; # Specify use of the inet.0 table to resolve routes rather
```

```
        # than the default policy, which uses the inet.3 table.
    }
    rib :voice.inet.0 { # Specify that the routing table for the voice topology for IPv4
        # unicast traffic be modified.
        resolution-ribs [ inet.3 :voice.inet.0 ]; # Specify that routes be resolved
        # using the inet.3 routing table for the voice topology.
    }
}
```

CHAPTER 3

Multitopology Routing Configuration Examples

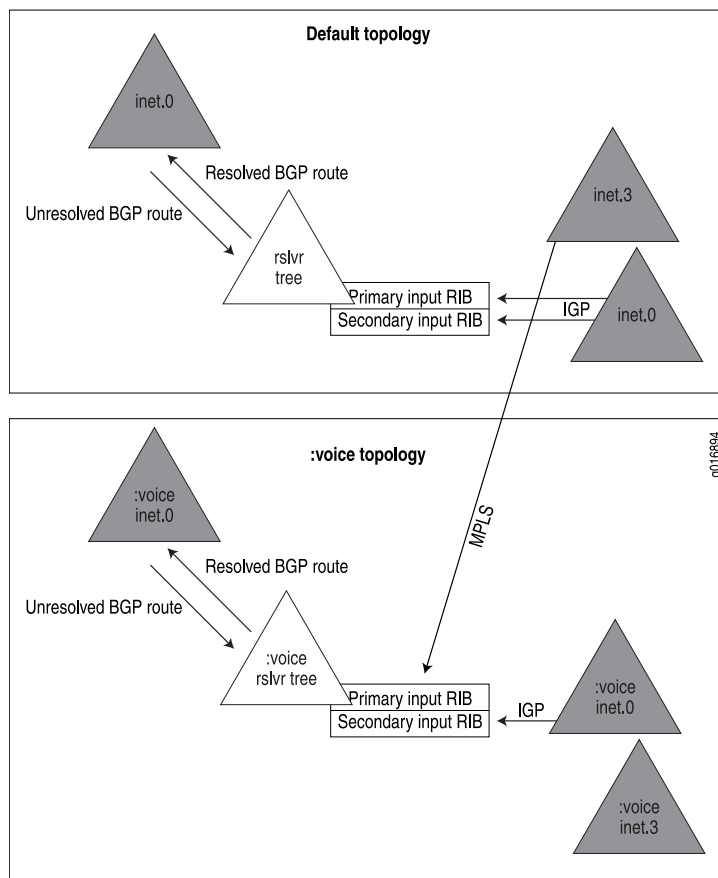
This chapter covers these topics:

- Example: Multitopology Routing Configuration on page 11
- For More Information on page 15

Example: Multitopology Routing Configuration

In this example, an MPLS network is running RSVP label-switched paths (LSPs) in the core. The network carries both best effort (BE) and expedited forwarding (EF) traffic, and the same destination prefixes are used for both types of traffic. A voice topology is created to enable the network to send EF traffic over the LSPs, but permit BE traffic to traverse the IP path. The voice, or EF traffic, is placed in the voice topology. The data, or BE traffic, is placed in the default topology. The voice and default topologies each create separate routing tables for storing routes, and each routing table creates a separate forwarding table. For each destination prefix, a different route is added to each topology-specific routing table. These routes are BGP routes. You configure filter-based forwarding so that the destination lookup is done in the two topology-specific routing tables based on the DSCP marking of the incoming packet. As Figure 4 on page 12 shows, MTR thus enables you to build two routes to the destination prefixes, one using LSP next hops for voice traffic and one using IP next hops for data traffic. The protocol next hop is resolved differently for each type of traffic. For voice EF traffic, routes from the **inet.3** routing table are taken into account, and for BE data traffic, only routes from the **inet.0** routing table are taken into account.

Figure 4: Route Resolution in Multitopology Routing



Configure the interfaces:

```
[edit]
interfaces {
  so-0/0/1 {
    unit 0 {
      family inet {
        address 1.13.1.1/24
      }
      family iso;
      family mpls;
    }
  }
  fe-2/2/0 {
    unit 0 {
      family inet {
        address 1.12.1.1/24
      }
      family iso;
      family mpls;
    }
  }
  fe/2/2/1 {
```

```

unit 0 {
  family inet {
    filter {
      input topo_selection; # Apply a firewall filter on the ingress interface.
      # This filter performs filter-based forwarding for the voice topology.
    }
    family iso;
    family mpls;
  }
}

```

Configure the voice topology. Configure a route resolution policy so that IPv4 routes for data traffic are resolved through the **inet.0** routing table, which functions as the routing table for the default topology. Configure a route resolution policy so that routes for voice traffic are resolved through the routing table for the voice topology (**:voice.inet.0**) and the MPLS routing table (**inet.3**).

```

[edit]
routing-options {
  autonomous-system 65300;
  resolution {
    rib inet.0 {
      resolution-ribs inet.0; # Specify use of the inet.0 routing table to resolve
      # IPv4 data traffic. This action prevents this traffic from being resolved using
      # the MPLS routing table (inet.3).
    }
    rib :voice.inet.0 {
      resolution-ribs [ inet.3 :voice.inet.0 ]; # Specify use of the MPLS routing
      # table (inet.3) and the routing table for the voice topology (:voice.inet.0)
      # to resolve IPv4 voice traffic. This action prevents voice traffic from being
      # resolved using the inet.0 routing table.
    }
  }
}
topologies {
  family inet {
    topology voice;
  }
}

```

Configure MPLS using RSVP label-switched paths. Configure BGP so that routes learned through BGP are installed in the appropriate routing table. In this example, the **topology** statement is used to install BGP routes for voice traffic into the routing table for the voice topology (**:voice.inet.0**). This action overrides the default behavior to resolve BGP routes only through the **inet.0** or **inet.3** routing tables. Configure an interior gateway protocol (IGP). In this example, you configure OSPF.

```

[edit]
protocols {
  rsvp {
    interface all;
    interface fxp0.0 {
      disable;
    }
  }
}

```

```
}
mpls {
  label-switched-path to_r3 {
    to 10.255.14.222;
    primary test;
  }
  path test {
    1.12.1.2 strict;
  }
  interface all;
}
bgp {
  group int {
    type internal;
    local-address 10.255.14.223;
    family inet {
      unicast {
        topology voice {
          community 70:1;
        }
      }
    }
  }
  neighbor 10.255.14.220;
  neighbor 10.255.14.218;
  neighbor 10.255.14.222;
}
ospf {
  topology voice topology-id 32;
  traffic-engineering;
  area 0.0.0.0 {
    interface lo0.0 {
      passive;
    }
    interface all;
    interface fxp0.0 {
      disable;
    }
    interface fe-2/2/1.0 {
      metric 1;
    }
    interface fe-2/2/0.0 {
      metric 1;
    }
    interface so-0/0/1.0 {
      metric 1;
    }
  }
}
}
```

Configure a class-of-service classifier on the ingress interface. In this example, the classifier type is **inet-precedence**, which evaluates incoming IPv4 packets and requires only the upper three bits of the DSCP field.

[edit]


```

class-of-service {
  interfaces {
    fe-2/2/1 {
      unit 0 {
        classifiers {
          inet-precedence default;
        }
      }
    }
  }
}

```

Configure filter-based forwarding. This filter is applied to the ingress interface. Traffic marked for expedited forwarding is forwarded to the routing table for the voice topology. All other traffic is forwarded to the routing table for the default topology.

```

[edit]
firewall {
  family inet {
    filter topo_selection {
      term ef {
        from {
          forwarding-class expedited-forwarding;
        }
        then {
          topology voice; # Forward expedited-forwarding traffic to the routing
                          # table for the voice topology (:voice.inet.0).
          accept;
        }
      }
      term default {
        then accept; # Forward all other traffic to the routing table for the default
                    # topology (inet.0).
      }
    }
  }
}

```

Verifying Your Work

To verify proper operation of Multitopology Routing, use the following commands:

- **show route summary**
- **show route table *routing-table-name***
- **show route rib-groups**

For More Information

For additional information about Multitopology Routing, see the following resources:

- *Junos OS Routing Protocols Configuration Guide*
- RFC 4915, *Multi-Topology (MT) Routing in OSPF*

PART 2

Index

- Index on page 19

Index

B

BGP in Multitopology Routing.....	4
configuring.....	8

F

filter-based forwarding in Multitopology Routing	
configuring.....	7

M

Multitopology Routing	
configuration examples.....	11
overview.....	3
system requirements.....	5

O

OSPF in Multitopology Routing.....	3
configuring.....	8

R

route resolution policy in Multitopology Routing	
configuring.....	9

S

static routes in Multitopology Routing	
configuring.....	9

T

Topologies in Multitopology Routing	
configuring.....	7

