7705 Service Aggregation Router Gen 2

Release 26.3.R1

Multicast Routing Protocols Guide

3HE 29564 AAAA TQZZA 01
Edition: 01
March 2026

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

Table of contents

List of tables

List of figures

1 Getting started

1.1 About this guide

This guide describes multicast routing protocols, troubleshooting, and proprietary entities and presents configuration and implementation examples.

This guide is organized into functional chapters and provides concepts and descriptions of the implementation flow, as well as Command Line Interface (CLI) syntax and command usage.

Unless otherwise indicated, the topics and commands described in this guide apply only to the 7705 SAR Gen 2 platforms listed in Platforms and terminology.

Command outputs shown in this guide are examples only; actual displays may differ depending on supported functionality and user configuration.

Note: Unless otherwise indicated, CLI commands, contexts, and configuration examples in this guide apply for both the classic CLI and the MD-CLI.

The SR OS CLI trees and command descriptions can be found in the following guides:

- *7705 SAR Gen 2 Classic CLI Command Reference Guide*
- *7705 SAR Gen 2 Clear, Monitor, Show, Tools CLI Command Reference Guide* (for both the MD-CLI and classic CLI)
- *7705 SAR Gen 2 MD-CLI Command Reference Guide*

Note: This guide generically covers Release 26.*x*.R*x* content and may contain some content that will be released in later maintenance loads. See the *SR OS R26.x.Rx Software Release Notes*, part number 3HE 29176 000*x* TQZZA, for information about features supported in each load of the Release 26.*x*.R*x* software. For a list of features and CLI commands that are present in SR OS but not supported on the 7705 SAR Gen 2 platforms, see "SR OS Features not Supported on SAR Gen 2" in the *SR OS R26.x.Rx Software Release Notes*.

1.2 Platforms and terminology

Note:
Unless explicitly noted otherwise, this guide uses the terminology defined in the following table to collectively designate the specified platforms.

Table 1: Platforms and terminology

Platform	Collective platform designation
7705 SAR-Hx	7705 SAR Gen 2
7705 SAR-Mx	

Platform	Collective platform designation
7705 SAR-1	

1.3 Conventions

This section describes the general conventions used in this guide.

1.3.1 Precautionary and information messages

The following information symbols are used in the documentation.

DANGER: Danger warns that the described activity or situation may result in serious personal injury or death. An electric shock hazard could exist. Before you begin work on this equipment, be aware of hazards involving electrical circuitry, be familiar with networking environments, and implement accident prevention procedures.

WARNING: Warning indicates that the described activity or situation may, or will, cause equipment damage, serious performance problems, or loss of data.

Caution: Caution indicates that the described activity or situation may reduce your component or system performance.

Note: Note provides additional operational information.

Tip: Tip provides suggestions for use or best practices.

1.3.2 Options or substeps in procedures and sequential workflows

Options in a procedure or a sequential workflow are indicated by a bulleted list. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform one of the listed options to complete the step.

Example: Options in a procedure

1. User must perform this step.
2. This step offers three options. User must perform one option to complete this step.
   - This is one option.
   - This is another option.
   - This is yet another option.

Substeps in a procedure or a sequential workflow are indicated by letters. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform two substeps (a. and b.) to complete the step.

Example: Substeps in a procedure

1. User must perform this step.
2. User must perform all substeps to complete this action.
   a. This is one substep.
   b. This is another substep.

Nested substeps within a procedure or a sequential workflow are indicated by roman numerals. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform two substeps (a. and b.) to complete the step. At substep b, the user must perform two additional substeps (i. and ii.) to complete the step.

Example: Nested substeps in a procedure

1. User must perform this step.
2. User must perform all substeps to complete this action.
   a. This is one substep.
   b. User must perform all nested substeps to complete this action.
      i. This is a nested substep.
      ii. This is another nested substep.

2 Introduction to multicast

This chapter provides information about multicast.

2.1 Multicast overview

IP multicast provides an effective method of many-to-many communication. With unicast, IP packets are sent from a single source to a single recipient. The source inserts the address of the target host in the IP header destination field of an IP datagram, and intermediate routers (if present) forward the datagram toward the target, in accordance with their respective routing tables.

Sometimes distribution needs individual IP packets be delivered to multiple destinations (such as audio or video streaming broadcasts). Multicast is a method of distributing datagrams sourced from one or more hosts to a set of receivers that may be distributed over different networks. This makes delivery of multicast datagrams significantly more complex.

Multicast sources can send a single copy of data using a single address for the entire group of recipients. The routers between the source and recipients forward the packets using the group address route. Multicast packets are delivered to a multicast group. A multicast group specifies a set of recipients who are interested in a data stream and are represented by an IP address from a specified range. Data addressed to the IP address is forwarded to the members of the group. A source host sends data to a multicast group by specifying the multicast group address in the destination IP address. A source does not have to register to send data to a group nor do they need to be a member of the group.

Routers and Layer 3 switches use IGMP to manage membership for a multicast session. When a host needs to receive one or more multicast session, it sends a join message for each multicast group it needs to join. When a host needs to leave a multicast group, it sends a leave message.

To extend multicast to the Internet, the multicast backbone (Mbone) is used. The Mbone is layered on top of portions of the Internet. These portions, or islands, are interconnected using tunnels. The tunnels allow multicast traffic to pass between the multicast-capable portions of the Internet. As more and more routers in the Internet are multicast-capable, the unicast and multicast routing table converges.

The original Mbone was based on Distance Vector Multicast Routing Protocol (DVMRP) and was very limited. However, the Mbone is converging around the following protocol set:

- IGMP
- Source-Specific Multicast Groups (SSM groups)
- Protocol Independent Multicast - Sparse Mode (PIM-SM)

2.2 Multicast models

This section describes the models which Nokia routers support to provide multicast.

2.2.1 ASM

Any-Source Multicast (ASM) is the IP multicast service model defined in RFC 1112, *Host Extensions for IP Multicasting*. An IP datagram is transmitted to a host group, a set of zero or more end-hosts identified by a single IP destination address (224.0.0.0 through 239.255.255.255 for IPv4). End-hosts can join and leave the group any time and there is no restriction on their location or number. This model supports multicast groups with arbitrarily many senders. Any end-host can transmit to a host group even if it is not a member of that group.

To combat the vast complexity and scaling issues that ASM represents, the IETF is developing a service model called Source Specific Multicast (SSM).

2.2.2 PIM-SSM

The Source Specific Multicast (SSM) service model defines a channel identified by an (S,G) pair, where S is a source address and G is an SSM destination address. In contrast to the ASM model, SSM only provides network-layer support for one-to-many delivery.

The SSM service model attempts to alleviate the following deployment problems that ASM has presented:

- **address allocation**

  SSM defines channels on a per-source basis. For example, the channel (S1,G) is distinct from the channel (S2,G), where S1 and S2 are source addresses, and G is an SSM destination address. This averts the problem of global allocation of SSM destination addresses and makes each source independently responsible for resolving address collisions for the various channels it creates.

- **access control**

  SSM provides an efficient solution to the access control problem. When a receiver subscribes to an (S,G) channel, it receives data sent only by the source S. In contrast, any host can transmit to an ASM host group. At the same time, when a sender picks an (S,G) channel to transmit on, it is automatically ensured that no other sender transmits on the same channel (except in the case of malicious acts such as address spoofing). This makes it harder to spam an SSM channel than an ASM multicast group.

- **handling of well-known sources**

  SSM requires only source-based forwarding trees, eliminating the need for a shared tree infrastructure. In terms of the IGMP, PIM-SM, MSDP, MBGP protocol suite, this implies that neither the RP-based shared tree infrastructure of PIM-SM nor the MSDP protocol is required. Thus, the complexity of the multicast routing infrastructure for SSM is low, making it viable for immediate deployment. MBGP is still required for distribution of multicast reachability information.

- anticipating that point-to-multipoint applications such as Internet TV will be significant in the future, the SSM model is better suited for such applications.

3 IGMP

This chapter provides information about IGMP.

3.1 IGMP overview

Internet Group Management Protocol (IGMP) is used by IPv4 hosts and routers to report their IP multicast group memberships to neighboring multicast routers. A multicast router keeps a list of multicast group memberships for each attached network, and a timer for each membership.

Multicast group memberships include at least one member of a multicast group on a specific attached network, not a list of all the members. With respect to each of its attached networks, a multicast router can assume one of two roles, querier or non-querier. There is normally only one querier per physical network.

A querier issues two types of queries, a general query and a group-specific query. General queries are issued to solicit membership information with regard to any multicast group. Group-specific queries are issued when a router receives a leave message from the node it perceives as the last group member remaining on that network segment.

Hosts wanting to receive a multicast session issue a multicast group membership report. These reports must be sent to all multicast enabled routers.

3.1.1 IGMP versions and interoperability requirements

If routers run different versions of IGMP, they negotiate the lowest common version of IGMP that is supported on their subnet and operate in that version.

- **Version 1**
  Specified in RFC 1112, *Host extensions for IP Multicasting*, was the first widely deployed version and the first version to become an Internet standard.
- **Version 2**
  Specified in RFC 2236, *Internet Group Management Protocol, Version 2*, added support for "low leave latency", that is, a reduction in the time it takes for a multicast router to learn that there are no longer any members of a particular group present on an attached network.
- **Version 3**
  Specified in RFC 3376, *Internet Group Management Protocol, Version 3*, adds support for source filtering; that is, the ability for a system to report interest in receiving packets only from specific source addresses, as required to support PIM-SSM, or from all but specific source addresses, sent to a particular multicast address.

  IGMPv3 must keep state per group per attached network. This group state consists of a filter-mode, a list of sources, and various timers. For each attached network running IGMP, a multicast router records the needed reception state for that network.

3.1.2 IGMP version transition

Nokia routers are capable of interpreting with routers and hosts running IGMPv1, IGMPv2, and/or IGMPv3. RFC 5186, *Internet Group Management Protocol Version 3 (IGMPv3)/Multicast Listener Discovery Version 2 (MLDv2) and Multicast Routing Protocol Interaction* describes some of the interoperability issues and how they affect the various routing protocols.

IGMPv3 (RFC 3376) specifies that if a router receives an earlier version query message on an interface, it must immediately switch into a compatibility mode with that earlier version. None of the previous versions of IGMP are source aware. Therefore, if this occurs, and the interface switches to Version 1 or 2 compatibility mode, any previously learned group memberships with specific sources (learned via the IGMPv3 specific include or exclude mechanisms) must be converted to non-source-specific group memberships. The routing protocol treats this as if there is no exclude definition present.

3.1.3 SSM groups

IGMPv3 allows a receiver to join a group and specify that it only wants to receive traffic for a group if that traffic comes from a particular source. If a receiver does this, and no other receiver on the LAN requires all the traffic for the group, then the designated router (DR) can omit performing a (*,G) join to set up the shared tree, and instead issue a source-specific (S,G) join only.

The range of multicast addresses from 232.0.0.0 to 232.255.255.255 is currently set aside for source-specific multicast in IPv4. For groups in this range, receivers should only issue source-specific IGMPv3 joins. If a PIM router receives a non-source-specific join for a group in this range, it should ignore it.

A Nokia router PIM router must silently ignore a received (*,G) PIM join message where G is a multicast group address from the multicast address group range that has been explicitly configured for SSM. This occurrence should generate an event. If configured, the IGMPv2 request can be translated into IGMPv3. The router allows for the conversion of an IGMPv2 (*,G) request into a IGMPv3 (S,G) request based on manual entries. A maximum of 32 SSM ranges is supported.

IGMPv3 also allows a receiver to join a group and specify that it only wants to receive traffic for a group if that traffic does not come from a specific source or sources. In this case, the DR performs a (*,G) join as normal, but can combine this with a prune for each of the sources the receiver does not want to receive.

3.1.4 Query messages

The IGMP query source address is configurable at two hierarchal levels. It can be configured globally at each router instance IGMP level and can be configured at individual at the group-interface level. The group-interface level overrides the src-ip address configured at the router instance level.

By default, subscribers with IGMP policies send IGMP queries with an all zero SRC IP address (0.0.0.0). However, some systems only accept and process IGMP query messages with non-zero SRC IP addresses. This feature allows the BNG to inter-operate with such systems.

3.2 Configuring IGMP with CLI

This section provides information to configure IGMP using the CLI.

3.2.1 IGMP configuration overview

The routers use IGMP to manage membership for a multicast session. IGMP is not enabled by default. When enabled, at least one interface must be specified in the IGMP context as IGMP is an interface function. Creating an interface enables IGMP. Traffic can only flow away from the router to an IGMP interface and to and from a PIM interface. A router directly connected to a source must have PIM enabled on the interface to that source. The traffic travels in a network from PIM interface to PIM interface and arrives finally on an IGMP enabled interface.

The IGMP CLI context allows you to specify an existing IP interface and modify the interface-specific parameters. Static IGMP group memberships can be configured to test multicast forwarding without a receiver host. When IGMP static group membership is enabled, data is forwarded to an interface without receiving membership reports from host members.

When static IGMP group entries on point-to-point links that connect routers to a rendezvous point (RP) are configured, the static IGMP group entries do not generate join messages toward the RP. When a host wants to receive multicast sessions it sends a join message for each multicast group it wants to join. Then, a leave message may be sent for each multicast group it no longer needs to participate with.

A multicast router keeps a list of multicast group memberships for each attached network, and an interval timer for each membership. Hosts issue a Multicast Group Membership Report when they want to receive a multicast session. The reports are sent to all multicast routers.

3.2.2 Basic IGMP configuration

About this task

Perform the following basic multicast configuration tasks:

Procedure

Step 1. Enable IGMP.

Step 2. Configure IGMP interfaces.

Step 3. Optional: Specify the IGMP version on the interface.

Step 4. Optional: Configure static (S,G)/(*,G).

Step 5. Optional: Configure SSM translation.

3.2.3 Configuring IGMP

3.2.3.1 Enabling IGMP

Use the commands in the following context to enable IGMP.

```
configure router igmp
```

The following example shows IGMP configuration information.

Example: MD-CLI

```
[ex:/configure router "base" igmp]
A:admin@node-2# info
    admin-state enable
    query-interval 125
    query-last-member-interval 1
    query-response-interval 10
    robust-count 2
    ...
```

Example: classic CLI

```
A:node-2>config>router# info
...
#------------------------------------------
echo "IGMP Configuration"
#------------------------------------------
        igmp
            query-interval 125
            query-last-member-interval 1
            query-response-interval 10
            robust-count 2
            no shutdown
        exit
#------------------------------------------
```

3.2.3.2 Configuring an IGMP interface

Use the following command to configure an interface for IGMP. You can reference interfaces configured in the router, IES service, or IES service video-interface context.

```
configure router igmp interface
```

The following example shows interfaces configured for IGMP.

Example: Configure interfaces for IGMP (MD-CLI)

```
[ex:/configure router "2" igmp]
A:admin@node-2# info
    ...
    interface "itf1" {
        admin-state enable
    }
    interface "itf2" {
        admin-state enable
    }
    interface "ip-1.1.1.3" {
        admin-state enable
    }
 ...
```

Example: Configure interfaces for IGMP (classic CLI)

```
A:node-2>config>router>igmp# info
----------------------------------------------
           ...
            interface "itf1"
```

```
                no shutdown
            exit
            interface "itf2"
                no shutdown
            exit
            interface "ip-1.1.1.3"
                no shutdown
            exit
            no shutdown
           ...
-----------------------------------------------
```

Use the commands in the **interface** context to configure the interface for IGMP. The following example shows some IGMP interface configuration options.

Example: Configure IGMP interface options (MD-CLI)

```
[ex:/configure router "2" igmp interface "itf1"]
A:admin@node-2# info
    admin-state enable
    maximum-number-group-sources 5
    maximum-number-sources 10
    version 2
    ...
```

Example: Configure IGMP interface options (classic CLI)

```
A:node-2>config>router>igmp# interface "itf1"
A:node-2>config>router>igmp>if# info
-----------------------------------------------
           version 2
           max-groups 5
           max-sources 10
           no shutdown
           ...
-----------------------------------------------
```

3.2.3.3 Configuring IGMP static multicast

About this task

This task describes how to configure an IGMP static multicast group and add a source IP address or starg entry. Use the commands in the following context to configure an IGMP static multicast group.

```
configure router igmp interface static
```

Procedure

Step 1. Configure an IGMP static multicast group.

```
configure router igmp interface ip-int-name static group ip-address
```

Step 2. Configure a source IP address or a static (*,G) entry for the group.

```
configure router igmp interface ip-int-name static group ip-address source ip-address
configure router igmp interface ip-int-name static group ip-address starg
```

Example

MD-CLI

```
[ex:/configure router "Base" igmp]
A:admin@node-2# info
    ...
    interface "itf1" {
        ...
        static {
            group 239.255.0.2 {
                source 172.22.184.197 { }
            }
        }
    }
    interface "itf2" {
        static {
            group 239.1.1.1 {
                starg
            }
        }
    }
```

Example

classic CLI

```
A:node-2>config>router>igmp# info
----------------------------------------------
        ...
        interface "itf1"
            ...
            static
                group 239.255.0.2
                    source 172.22.184.197
                exit
            exit
        exit
        interface "itf2"
            ...
            static
                group 239.1.1.1
                    starg
                exit
            exit
        exit
----------------------------------------------
```

3.2.3.4 Configuring SSM translation

Procedure

Use the commands in the following context to configure SSM translation for IGMP.

```
configure router igmp ssm-translate
```

The following example shows an SSM translation configuration for IGMP.

Example: MD-CLI

```
[ex:/configure router "Base" igmp]
A:admin@node-2# info
    ...
    ssm-translate {
        group-range start 239.255.0.1 end 239.2.2.2 {
            source 10.1.1.1 { }
        }
    }
    ...
```

Example: classic CLI

```
A:node-2>config>router>igmp# info
----------------------------------------------
        ...
        ssm-translate
            grp-range 239.255.0.1 239.2.2.2
                source 10.1.1.1
            exit
        exit
        ...
----------------------------------------------
```

3.2.4 Disabling IGMP

IGMP is enabled by default. Use the following command to disable IGMP:

- **MD-CLI**

```
configure router igmp admin-state disable
```

- **classic CLI**

```
configure router igmp shutdown
```

4 MLD

This chapter provides information about Multicast Listener Discovery (MLD).

4.1 MLD overview

Multicast Listener Discovery (MLD) is the IPv6 version of IGMP and belongs to the Source Specific Multicast (SSM) service model (see IPv6 PIM models for more information). The purpose of MLD is to allow each IPv6 router to discover the presence of multicast listeners on its directly attached links, and to discover specifically which multicast groups are of interest to those neighboring nodes.

MLD is a sub-protocol of ICMPv6. MLD message types are a subset of the set of ICMPv6 messages, and MLD messages are identified in IPv6 packets by a preceding Next Header value of 58. All MLD messages are sent with a link-local IPv6 source address, a Hop Limit of 1, and an IPv6 Router Alert option in the Hop-by-Hop Options header.

4.1.1 MLDv1

Similar to IGMPv2, MLDv1 reports only include the multicast group addresses that listeners are interested in, and do not include the source addresses. To work with the PIM-SSM model, a similar SSM translation function is required when MLDv1 is used.

SSM translation allows an IGMPv2 device to join an SSM multicast network through the router that provides such a translation capability. Currently SSM translation can done at a box level, but this does not allow a per-interface translation to be specified. SSM translation per interface offers the ability to have a same (*,G) mapped to two different (S,G) on two different interfaces to provide flexibility.

4.1.2 MLDv2

MLDv2 is backward compatible with MLDv1 and adds the ability for a node to report interest in listening to packets with a particular multicast group only from specific source addresses or from all sources except for specific source addresses.

4.2 Configuring MLD with CLI

This section provides information about configuring MLD using the command line interface.

4.2.1 MLD configuration overview

The routers use MLD to manage membership for a multicast session. MLD is not enabled by default. Creating an interface enables MLD. When enabled, at least one interface must be specified in the **mld** context because MLD is an interface function. Traffic can only flow away from the router to an MLD

interface and to and from a PIM interface. A router directly connected to a source must have PIM enabled on the interface to the source. The traffic travels in a network from PIM interface to PIM interface and arrives finally on an MLD-enabled interface.

Use the MLD CLI to specify an existing IP interface and modify the interface-specific parameters. Static MLD group memberships can be configured to test multicast forwarding without a receiver host. If MLD static group membership is enabled, data is forwarded to an interface without receiving membership reports from host members.

If static MLD group entries on point-to-point links that connect routers to a rendezvous point (RP) are configured, the static MLD group entries do not generate join messages toward the RP. When a host wants to receive multicast sessions, the host sends a join message for each multicast group it wants to join. A leave message may be sent for each multicast group it no longer needs to participate with.

A multicast router keeps a list of multicast group memberships for each attached network and an interval timer for each membership. Hosts issue a multicast group membership report when they want to receive a multicast session. The reports are sent to all multicast routers.

4.2.2 Basic MLD configuration

Prerequisites

Perform the following basic multicast configuration tasks:

Procedure

Step 1. Enable MLD.

Step 2. Configure MLD interfaces.

Step 3. Optional: Specify the MLD version on the interface.

Step 4. Optional: Configure static (S,G)/(*,G).

Step 5. Optional: Configure SSM translation.

4.2.3 Configuring MLD

4.2.3.1 Enabling MLD

Use the commands in the following context to configure and enable MLD for the router.

```
configure router mld
```

The following example shows a basic configuration with MLD enabled.

Example: MD-CLI

```
[ex:/configure router "Base" mld]
A:admin@node-2# info
    ...
    admin-state enable
    group-if-query-source-address
    query-interval 125
    query-last-member-interval 1
```

```
    query-response-interval 10
    robust-count 2
    ...
```

Example: classic CLI

```
A:node-2>config>router>mld# info
----------------------------------------------
        ...
        no grp-if-query-src-ip
        query-interval 125
        query-last-listener-interval 1
        query-response-interval 10
        robust-count 2
        no shutdown
----------------------------------------------
```

4.2.3.2 Configuring MLD interfaces

Use the commands in the following context to configure MLD interfaces for the router.

```
configure router mld interface
```

The following example shows interfaces configured for MLD.

Example: MD-CLI

```
[ex:/configure router "2" mld]
A:admin@node-2# info
    ...
    interface "lax-sjc" {
        admin-state enable
    }
    interface "lax-vls" {
        admin-state enable
    }
    interface "pl-ix" {
        admin-state enable
    }
 ...
```

Example: classic CLI

```
A:node-2>config>router>mld# info
----------------------------------------------
           ...
            interface "lax-sjc"
                no shutdown
            exit
            interface "lax-vls"
                no shutdown
            exit
            interface "pl-ix"
                no shutdown
            exit
           ...
----------------------------------------------
```

4.2.3.3 Configuring MLD static multicast

About this task

This task describes how to configure a static multicast group with a source IP address or (*,G) entry for an MLD interface.

Procedure

Step 1. Use the following command to configure an MLD static multicast group for the router.

```
configure router mld interface static group grp-ipv6-address
```

Example

Static group configuration (MD-CLI)

```
[ex:/configure router "Base" mld]
A:admin@node-2# info
    ...
    interface "lax-vls" {
        static {
            group ff0e::db8:7 { }
        }
    }
    interface "lax-sjc" {
        static {
            group ff0e::db8:9 { }
        }
    }
    ...
```

Example

Static group configuration (classic CLI)

```
A:node-2>config>router>mld# info
----------------------------------------------
        ...
        interface "lax-vls"
            static
                group ff0e::db8:7
            exit
        exit
        interface "lax-sjc"
            static
                group ff0e::db8:9
            exit
        exit
        ...
----------------------------------------------
```

Step 2. Configure a source or (*,G) entry for the static multicast group.

a. Use the following command to configure a source for the static group.

```
configure router mld interface static group source ipv6-address
```

Example

Static group source configuration (MD-CLI)

```
[ex:/configure router "Base" mld]
A:admin@node-2# info
    ...
     interface "lax-vls" {
        static {
            group ff0e::db8:7 {
                source 2001:db8::1 { }
            }
        }
    }
    ...
```

Example

Static group source configuration (classic CLI)

```
A:node-2>config>router>mld# info
----------------------------------------------
        ...
        interface "lax-vls"
            static
                group ff0e::db8:7
                    source 2001:db8::1
                exit
            exit
        exit
        ...
----------------------------------------------
```

b. Use the following command to configure a (*,G) entry for the static group.

```
configure router mld interface static group starg
```

Example

Static group (*,G) entry configuration (MD-CLI)

```
[ex:/configure router "Base" mld]
A:admin@node-2# info
    ...
    interface "lax-sjc" {
        static {
            group ff0e::db8:9 {
                starg
            }
        }
    }
    ...
```

Example

Static group (*,G) entry configuration (classic CLI)

```
A:node-2>config>router>mld# info
----------------------------------------------
        ...
        interface "lax-sjc"
            static
```

```
                group ff0e::db8:9
                    starg
                exit
            exit
        exit
        ...
-----------------------------------------------
```

4.2.3.4 Configuring SSM translation

Use commands in the following context to configure SSM translation for MLD.

```
configure router mld ssm-translate
```

The following example displays the command usage to configure MLD for the router.

Note: The group range is not created until the source is specified.

Example: MD-CLI

```
[ex:/configure router "Base" mld]
A:admin@node-2# info
    ...
    ssm-translate {
        group-range start ff0e::db8:7 end ff0e::db8:9 {
            source 2001:db8::1 { }
        }
    }
    ...
```

Example: classic CLI

```
A:node-2>config>router>mld# info
----------------------------------------------
        ...
        ssm-translate
            grp-range ff0e::db8:7 ff0e::db8:9
                source 2001:db8::1
            exit
        exit
        ...
----------------------------------------------
```

4.2.4 Disabling MLD

MLD is enabled by default. Use the following command to disable MLD:

- **MD-CLI**

```
configure router mld admin-state disable
```

- **classic CLI**

```
configure router mld shutdown
```

5 PIM

This chapter provides information about PIM.

5.1 PIM-SM

PIM-SM leverages the unicast routing protocols that are used to create the unicast routing table, OSPF, IS-IS, BGP, and static routes. Because PIM uses this unicast routing information to perform the multicast forwarding function it is effectively IP protocol independent. Unlike DVMRP, PIM does not send multicast routing tables updates to its neighbors.

PIM-SM uses the unicast routing table to perform the Reverse Path Forwarding (RPF) check function instead of building up a completely independent multicast routing table.

PIM-SM only forwards data to network segments with active receivers that have explicitly requested the multicast group. PIM-SM in the ASM model initially uses a shared tree to distribute information about active sources. Depending on the configuration options, the traffic can remain on the shared tree or switch over to an optimized source distribution tree. As multicast traffic starts to flow down the shared tree, routers along the path determine if there is a better path to the source. If a more direct path exists, then the router closest to the receiver sends a join message toward the source and then reroutes the traffic along this path.

As stated above, PIM-SM relies on an underlying topology-gathering protocol to populate a routing table with routes. This routing table is called the Multicast Routing Information Base (MRIB). The routes in this table can be taken directly from the unicast routing table, or it can be different and provided by a separate routing protocol such as MBGP. Regardless of how it is created, the primary role of the MRIB in the PIM-SM protocol is to provide the next hop router along a multicast-capable path to each destination subnet. The MRIB is used to determine the next hop neighbor to whom any PIM join/prune message is sent. Data flows along the reverse path of the join messages. Thus, in contrast to the unicast RIB that specifies the next hop that a data packet would take to get to some subnet, the MRIB gives reverse-path information, and indicates the path that a multicast data packet would take from its origin subnet to the router that has the MRIB.

Note: For correct functioning of the PIM protocol, multicast data packets need to be received by the CPM CPU. Therefore, CPM filters and management access filters must be configured to allow forwarding of multicast data packets.

5.1.1 PIM-SM functions

PIM-SM functions have three phases.

5.1.1.1 Phase one

In this phase, a multicast receiver expresses its interest in receiving traffic destined for a multicast group. Typically it does this using IGMP, but other mechanisms may also serve this purpose. One of the local routers of the receiver is elected as the designated router (DR) for that subnet. When the expression of interest is received, the DR sends a PIM join message toward the RP for that multicast group. This join

message is known as a (*,G) join because it joins group G for all sources to that group. The (*,G) join travels hop-by-hop toward the RP for the group, and in each router it passes through, the multicast tree state for group G is instantiated.

Eventually, the (*,G) join either reaches the RP or reaches a router that already has the (*,G) join state for that group. When many receivers join the group, their join messages converge on the RP and form a distribution tree for group G that is rooted at the RP. The distribution tree is called the RP tree or the shared tree (because it is shared by all sources sending to that group). Join messages are re-sent periodically as long as the receiver remains in the group. When all receivers on a leaf network leave the group, the DR sends a PIM (*,G) prune message toward the RP for that multicast group. However, if the prune message is not sent for any reason, the state eventually times out.

A multicast data sender starts sending data destined for a multicast group. The local router of the sender (the DR) takes these data packets, unicast-encapsulates them, and sends them directly to the RP. The RP receives these encapsulated data packets, removes the encapsulation, and forwards them to the shared tree. The packets then follow the (*,G) multicast tree state in the routers on the RP tree, and are replicated wherever the RP tree branches, and eventually reach all the receivers for that multicast group. The process of encapsulating data packets to the RP is called registering, and the encapsulation packets are known as PIM register packets.

At the end of phase one, multicast traffic is flowing encapsulated to the RP, and then natively over the RP tree to the multicast receivers.

5.1.1.2 Phase two

In this phase, register-encapsulation of data packets is performed. However, register-encapsulation of data packets is inefficient for the following reasons:

- Encapsulation and de-encapsulation can be resource-intensive operations for a router to perform, depending on whether the router has appropriate hardware for the tasks.
- Traveling to the RP and then back down the shared tree can cause the packets to travel a relatively long distance to reach receivers that are close to the sender. For some applications, increased latency is unwanted.

Although register-encapsulation can continue indefinitely, for the previous reasons, the RP normally switches to native forwarding. To do this, when the RP receives a register-encapsulated data packet from source S on group G, it normally initiates an (S,G) source-specific join toward S. This join message travels hop-by-hop toward S, instantiating an (S,G) multicast tree state in the routers along the path.

The (S,G) multicast tree state is used only to forward packets for group G if those packets come from source S. Eventually, the join message reaches the S subnet or a router that already has the (S,G) multicast tree state, and packets from S start to flow following the (S,G) tree state toward the RP. These data packets can also reach routers with a (*,G) state along the path toward the RP, and if this occurs, they take a shortcut to the RP tree at this point.

While the RP is in the process of joining the source-specific tree for S, the data packets continue being encapsulated to the RP. When packets from S also start to arrive natively at the RP, the RP receives two copies of each of these packets. At this point, the RP starts to discard the encapsulated copy of these packets and sends a register-stop message back to the DR of S to prevent the DR from unnecessarily encapsulating the packets. At the end of phase two, traffic is flowing natively from S along a source-specific tree to the RP and from there along the shared tree to the receivers. Where the two trees intersect, traffic can transfer from the shared RP tree to the shorter source tree.

Note: A sender can start sending before or after a receiver joins the group, therefore phase two may occur before the shared tree to the receiver is built.

5.1.1.3 Phase three

In this phase, the RP joins back toward the source using the shortest path tree (SPT). Although having the RP join back toward the source removes the encapsulation overhead, it does not completely optimize the forwarding paths. For many receivers, the route via the RP can involve a significant detour when compared with the shortest path from the source to the receiver.

To obtain lower latencies, a router on the LAN of the receiver, typically the DR, may optionally initiate a transfer from the shared tree to a source-specific SPT. To do this, it issues an (S,G) join toward S. This instantiates the (S,G) state in the routers along the path to S. Eventually, this join either reaches the S subnet or reaches a router that already has the (S,G) state. When this happens, data packets from S flow following the (S,G) state until they reach the receiver.

At this point, the receiver (or a router upstream of the receiver) is receiving two copies of the data—one from the SPT, and one from the RP tree. When the first traffic starts to arrive from the SPT, the DR or upstream router starts to drop the packets for G from S that arrive via the RP tree. In addition, it sends an (S,G) prune message toward the RP. The prune message travels hop-by-hop, instantiating an (S,G) state along the path toward the RP, indicating that traffic from S for G should not be forwarded in this direction. The prune message is propagated until it reaches the RP or a router that still needs the traffic from S for other receivers.

By now, the receiver is receiving traffic from S along the SPT between the receiver and S. In addition, the RP is receiving the traffic from S, but this traffic is no longer reaching the receiver along the RP tree. As far as the receiver is concerned, this is the final distribution tree.

5.1.2 Encapsulating data packets in the register tunnel

Conceptually, the register tunnel is an interface with a smaller MTU than the underlying IP interface toward the RP. IP fragmentation on packets forwarded on the register tunnel is performed based on this smaller MTU. The encapsulating DR can perform path-MTU discovery to the RP to determine the effective MTU of the tunnel. This smaller MTU takes both the outer IP header and the PIM register header overhead into consideration.

5.1.3 PIM bootstrap router mechanism

For correct operation, every PIM-SM router within a PIM domain must be able to map a particular global-scope multicast group address to the same RP. If this is not possible, black holes can appear (this is where some receivers in the domain cannot receive some groups). A domain in this context is a contiguous set of routers that all implement PIM and are configured to operate within a common boundary.

The Bootstrap Router (BSR) mechanism provides a way in which viable group-to-RP mappings can be created and distributed to all the PIM-SM routers in a domain. Each candidate BSR originates Bootstrap Messages (BSMs). Every BSM contains a BSR priority field. Routers within the domain flood the BSMs throughout the domain. A candidate BSR that hears about a higher-priority candidate BSR suppresses sending more BSMs for a period of time. The single remaining candidate BSR becomes the elected BSR, and its BSMs inform the other routers in the domain that it is the elected BSR.

The PIM bootstrap routing mechanism is adaptive, meaning that if an RP becomes unreachable, the event is detected and the mapping tables are modified so that the unreachable RP is no longer used, and the new tables are rapidly distributed throughout the domain.

5.1.4 PIM-SM routing policies

Multicast traffic can be restricted from specific source addresses by creating routing policies. Join messages can be filtered using import filters. PIM join policies can be used to reduce denial of service attacks and subsequent PIM state explosion in the router and to remove unwanted multicast streams at the edge of the network before it is carried across the core.

Use commands in the following context to configure route policies:

- **MD-CLI**

```
configure policy-options
```

- **classic CLI**

```
configure router policy-options
```

Join and register route policy match criteria for PIM-SM can specify the following:

- router interface or interfaces specified by name or IP address
- neighbor address (the source address in the IP header of the join and prune message)
- multicast group address embedded in the join and prune message
- multicast source address embedded in the join and prune message

Join policies can be used to filter PIM join messages so no (*,G) or (S,G) state is created on the router.

The following table lists the join filter policy match conditions.

Table 2: Join filter policy match conditions

Match condition	Matches
Interface	RTR interface by name
Neighbor	The neighbors source address in the IP header
Group Address	Multicast Group address in the join/prune message
Source Address	Source address in the join/prune message

PIM register message are sent by the first hop designated router that has a direct connection to the source. This serves a dual purpose:

- notifies the RP that a source has active data for the group
- delivers the multicast stream in register encapsulation to the RP and its potential receivers
- if no one has joined the group at the RP, the RP ignores the registers

In an environment where the sources to particular multicast groups are always known, it is possible to apply register filters at the RP to prevent any unwanted sources from transmitting multicast stream. You

can apply these filters at the edge so that register data does not travel unnecessarily over the network toward the RP.

The following table lists the register filter policy match conditions.

Table 3: Register filter policy match conditions

Match condition	Matches
Interface	RTR interface by name
Group Address	Multicast Group address in the join/prune message
Source Address	Source address in the join/prune message

5.1.5 RPF checks

Multicast implements a Reverse Path Forwarding (RPF) check. RPF checks the path that multicast packets take between their sources and the destinations to prevent loops. Multicast requires that an incoming interface be the outgoing interface used by unicast routing to reach the source of the multicast packet. RPF forwards a multicast packet only if it is received on an interface that is used by the router to route to the source.

If the forwarding paths are modified because of routing topology changes, any dynamic filters that may have been applied must be re-evaluated. If filters are removed, the associated alarms are also cleared.

5.1.6 Distributing PIM joins over multiple ECMP paths

The per bandwidth/round robin method is commonly used for multicast load-balancing, but the interface in an ECMP set can also be used for a specific channel to be predictable without knowledge of other channels that use the ECMP set.

Use the following command to distribute PIM joins over multiple ECMP paths based on a hash of S and G:

- **MD-CLI**

```
configure router pim mc-ecmp-hashing
```

- **classic CLI**

```
configure router pim mc-ecmp-hashing-enabled
```

When a link in the ECMP set is removed, multicast streams that use this link are redistributed over the remaining ECMP links using the same hash algorithm. When a link is added to the ECMP set, new joins may be allocated to the new link based on the hash algorithm. Existing multicast streams using the other ECMP links stay on those links until they are pruned, unless the **rebalance** command option is specified.

The default is not enabled, which means that the use of multiple ECMP paths (if enabled at the **configure service vprn** context) is controlled through the existing implementation and the **mc-ecmp-balance** command.

Note: You cannot use the **mc-ecmp-balance** command when MC ECMP hashing is enabled in the same context.

To achieve distribution of streams across the ECMP links, the hashing steps are as follows:

1. For a specific (S,G) get all possible next hops.
2. Sort these next hops based on next hop address.
3. XOR S and G addresses.
4. Hash the XORed address over the number of PIM next hops.
5. Use the hash value obtained in step 4, and set that element in the sorted list that was obtained in step 2 as the preferred next hop
6. If this element is not available or is not a PIM next hop (PIM neighbor), the next available next hop is chosen.

Use the following command to display the PIM status for the router instance.

```
show router 100 pim status
```

The following example displays the PIM status indicating ECMP hashing is disabled.

Output example: PIM status indicating ECMP hashing is disabled

```
===============================================================================
PIM Status ipv4
===============================================================================
Admin State                       : Up
Oper State                        : Up

IPv4 Admin State                  : Up
IPv4 Oper State                   : Up

BSR State                         : Accept Any

Elected BSR
    Address                       : None
    Expiry Time                   : N/A
    Priority                      : N/A
    Hash Mask Length              : 30
    Up Time                       : N/A
    RPF Intf towards E-BSR        : N/A

Candidate BSR
    Admin State                   : Down
    Oper State                    : Down
    Address                       : None
    Priority                      : 0
    Hash Mask Length              : 30

Candidate RP
    Admin State                   : Down
    Oper State                    : Down
    Address                       : 0.0.0.0
    Priority                      : 192
    Holdtime                      : 150

SSM-Default-Range                 : Enabled
SSM-Group-Range
    None
```

```
MC-ECMP-Hashing                   : Disabled

Policy                            : None

RPF Table                         : rtable-u

Non-DR-Attract-Traffic            : Disabled
===============================================================================
```

Use commands in the following context to configure PIM.

```
configure service vprn pim
```

PIM configuration includes:

- group-prefix shortest-path switchover thresholds
- interfaces
- import policies
- MC-ECMP traffic balancing or hash-based multicast balancing over ECMP links
- RP
- SSM group ranges

Example: PIM configuration for a VPRN service (MD-CLI)

```
[ex:/configure service vprn "5" pim]
A:admin@node-2# info
    admin-state enable
    apply-to all
    mc-ecmp-balance false
    mc-ecmp-hashing {
        rebalance true
    }
    rp {
        ipv4 {
            bsr-candidate {
                admin-state disable
            }
            static {
                address 10.3.3.3 {
                    group-prefix 224.0.0.0/4 { }
                }
            }
        }
            rp-candidate {
                admin-state disable
            }
        }
    }
```

Example: PIM configuration for a VPRN service (classic CLI)

```
-----------------------------------------------
A:node-2>config>service>vprn>pim# info
-----------------------------------------------
                apply-to all
                rp
                    static
```

```
                        address 10.3.3.3
                            group-prefix 224.0.0.0/4
                        exit
                    exit
                    bsr-candidate
                        shutdown
                    exit
                    rp-candidate
                        shutdown
                    exit
                exit
                no mc-ecmp-balance
                mc-ecmp-hashing-enabled
                no shutdown
----------------------------------------------
```

Use the following command to show distribution of PIM joins over multiple ECMP paths for the specified router instance.

```
show router 100 pim group
```

Output example: Distribution of PIM joins over multiple ECMP paths

```
===============================================================================
PIM Groups ipv4
===============================================================================
Group Address                          Type     Spt Bit Inc Intf       No.Oifs
   Source Address                         RP
-------------------------------------------------------------------------------
239.1.1.1                              (S,G)    spt     to_C0          1
   172.0.100.33                           10.20.1.6
239.1.1.2                              (S,G)    spt     to_C3          1
   172.0.100.33                           10.20.1.6
239.1.1.3                              (S,G)    spt     to_C2          1
   172.0.100.33                           10.20.1.6
239.1.1.4                              (S,G)    spt     to_C1          1
   172.0.100.33                           10.20.1.6
239.1.1.5                              (S,G)    spt     to_C0          1
   172.0.100.33                           10.20.1.6
239.1.1.6                              (S,G)    spt     to_C3          1
   172.0.100.33                           10.20.1.6

239.2.1.1                              (S,G)    spt     to_C0          1
   172.0.100.33                           10.20.1.6
239.2.1.2                              (S,G)    spt     to_C3          1
   172.0.100.33                           10.20.1.6
239.2.1.3                              (S,G)    spt     to_C2          1
   172.0.100.33                           10.20.1.6
239.2.1.4                              (S,G)    spt     to_C1          1
   172.0.100.33                           10.20.1.6
239.2.1.5                              (S,G)    spt     to_C0          1
   172.0.100.33                           10.20.1.6
239.2.1.6                              (S,G)    spt     to_C3          1
   172.0.100.33                           10.20.1.6

239.3.1.1                              (S,G)    spt     to_C0          1
   172.0.100.33                           10.20.1.6
239.3.1.2                              (S,G)    spt     to_C3          1
   172.0.100.33                           10.20.1.6
239.3.1.3                              (S,G)    spt     to_C2          1
   172.0.100.33                           10.20.1.6
239.3.1.4                              (S,G)    spt     to_C1          1
```

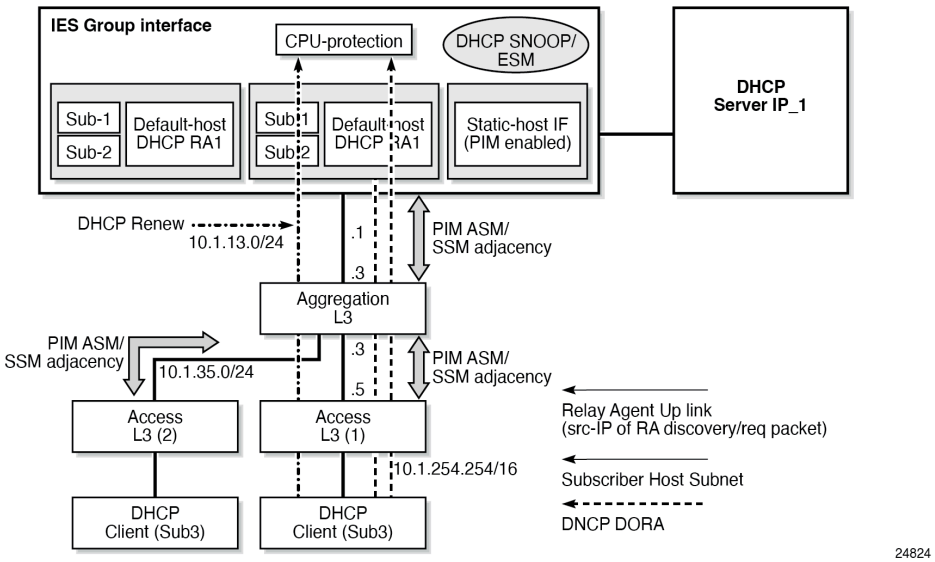
```
   172.0.100.33                            10.20.1.6
239.3.1.5                               (S,G)    spt     to_C0          1
   172.0.100.33                            10.20.1.6
239.3.1.6                               (S,G)    spt     to_C3          1
   172.0.100.33                            10.20.1.6

239.4.1.1                               (S,G)    spt     to_C0          1
   172.0.100.33                            10.20.1.6
239.4.1.2                               (S,G)    spt     to_C3          1
   172.0.100.33                            10.20.1.6
239.4.1.3                               (S,G)    spt     to_C2          1
   172.0.100.33                            10.20.1.6
239.4.1.4                               (S,G)    spt     to_C1          1
   172.0.100.33                            10.20.1.6
239.4.1.5                               (S,G)    spt     to_C0          1
   172.0.100.33                            10.20.1.6
239.4.1.6                               (S,G)    spt     to_C3          1
   172.0.100.33                            10.20.1.6
-------------------------------------------------------------------------------
Groups : 24
===============================================================================
```

5.1.7 PIM interface on IES subscriber group interfaces

PIM on a subscriber group interface allows for SAP-level replication over an ESM Group interface by establishing PIM adjacency to a downstream router. The following figure depicts the model.

Figure 1: PIM interface on IES subscriber group interface



On an IES subscriber-interface, an Ethernet SAP is configured (LAG or physical port). On the SAP, a static-host is configured for connectivity to downstream Layer 3 aggregation devices (including PIM adjacency) while multiple default-hosts can be configured for subscriber traffic. Single SAP with a single static-host per group interface is supported to establish PIM adjacency on a subscriber group interface. Both IPv4 PIM ASM and SSM are supported.

Feature restrictions:

- Only IPv4 PIM is supported with a single static host used to form a PIM interface under a group interface. Using multiple hosts or non-static hosts is not supported. Configuring IPv6 in the following context is not blocked, but takes no effect.

  ```
  configure router pim interface
  ```

- The following command does not apply to PIM interfaces on IES subscriber group interfaces.

  ```
  configure router pim apply-to
  ```

- PIM on group interfaces is not supported in VPRN context.
- Extranet is not supported.
- Locally attached receivers are not supported (no IGMP or MLD and PIM mix in OIF list).
- Default anti-spoofing must be configured (IP+MAC).
- A subscriber profile with a PIM policy enabled (**configure subscriber-mgmt sub-profile**) cannot combine with the following policies:
  - **host tracking policy**
    This option applies a host tracking policy.
  - **IGMP policy**
    This option applies an IGMP policy.
  - **MLD policy**
    This option applies an MLD policy.
  - **NAT policy**
    This option applies a NAT policy.
  - **Subscriber MCAC policy**
    This option applies a subscriber MCAC policy that can be used when configured in PIM interface context.

5.1.8 VRRP aware PIM

The Virtual Router Redundancy Protocol (VRRP) eliminates the single point of failure inherent in the static default-routed environment. VRRP describes a method of implementing a redundant IP interface that provides dynamic failover if the VRRP master router (MR) becomes unavailable.

VRRP provides information about the state of a router. However, PIM operates independently of VRRP group states. The PIM DR and the VRRP MR may not be the same router and IP multicast traffic may not necessarily follow the same path as elected by VRRP.

To leverage the redundancy capabilities of VRRP that are lacking in PIM, the VRRP Aware PIM mechanism allows PIM to monitor and react to changes in the VRRP MR. This ensures that the multicast traffic follows the unicast traffic through the same gateway as the VRRP MR, providing consistent IP multicast forwarding in a redundant network.

5.1.8.1 Configuring VRRP aware PIM

The VRRP Aware PIM feature enables PIM to track the state of a VRRP instance and to identify whether the associated VRRP interface is the master. PIM uses an operational group option (**oper-group** *group-*

name) to monitor the state of VRRP. One operational group can be created for IPv4, and another for IPv6. When VRRP is the MR, the operational group is up; for all other VRRP states, the operational group is down. A VRRP instance can only be associated with one operational group, and an operational group can have one or more associated VRRP instances. This feature is supported on base router, IES, and VPRN interfaces.

If the monitored interface is the VRRP MR, PIM becomes the DR by setting its priority to the configured oper-group active-priority value. For the router to become the DR, the correct priorities must be configured so the active priority of the **oper-group** is the highest priority on the IP interface.

If a PIM router is the DR and then receives an indication from VRRP that the interface is no longer the VRRP MR, PIM relinquishes the DR role by setting its priority back to the default or configured priority value.

If the configured VRRP instance or **oper-group** is not configured, PIM operates as normal with the default or configured priority value. A change in the operational group status is independent of the address family; IPv4 and IPv6 priorities are configured independently of each other. Two operational groups are supported per PIM interface, one for IPv4 and one for IPv6.

5.1.8.2 Guidelines for configuring VRRP Aware PIM

When configuring VRRP Aware PIM, consider the following recommendations:

- Configure VRRP to use BFD to speed up failure detection in addition to the functionality provided by VRRP Aware PIM.
- To optimize failover, enable the following command on the primary and secondary routers to make them a hot-standby redundant pair.

```
configure router pim non-dr-attract-traffic
```

Note: This configuration ignores the DR state and attracts traffic to populate the router's PIM database. Do not use this configuration if multicast traffic must only follow the VRRP MR.

- Configure the group **up** time on the primary router and the group **down** time on the secondary router to the time needed to repopulate the PIM database; for example, 10 seconds. This allows the primary router to populate its PIM database again before becoming the DR and recover from the secondary back to the primary router if a failure occurs from the primary to secondary router. Use the following commands to configure the **up** and **down** times.

```
configure service oper-group hold-time group up
configure service oper-group hold-time group down
```

  Configure the **up** time on the secondary router to 0, so that it assumes the DR role immediately if the primary router fails. The **up** hold time is set to 4 seconds by default, which delays the DR change unnecessarily.

- Sticky DR enables the secondary router to continue to act as the DR after the primary router comes back up. Sticky DR is incompatible with the VRRP Aware PIM mechanism that tracks the VRRP MR. You should disable it if it is configured with the following command.

```
configure router pim interface sticky-dr
```

The following example shows a basic configuration for VRRP Aware PIM.

Example: MD-CLI

```
[ex:/configure service]
A:admin@node-2# info
    interface "to-lan" {
        ipv4 {
            vrrp 1 {
                oper-group "VAwP1"
            }
        }
    }
    interface "to-lan2" {
        ipv4 {
            vrrp 1 {
                oper-group "VAwP2"
            }
        }
    }
    oper-group "VAwP1" {
    }
    oper-group "VAwP2" (
    }
    vprn "1" {
        customer "1"
        pim {
            interface "to-lan" {
                ipv4 {
                    monitor-oper-group {
                        name "VAwP1"
                        operation add
                        priority-delta 90
                    }
                }
                ipv6 {
                    monitor-oper-group {
                        name "VAwP1"
                        operation add
                        priority-delta 90
                    }
                }
            }
            interface "to-lan2" {
                ipv4 {
                    monitor-oper-group {
                        name "VAwP2"
                        operation add
                        priority-delta 90
                    }
                }
                ipv6 {
                    monitor-oper-group {
                        name "VAwP2"
                        operation add
                        priority-delta 90
                    }
                }
            }
        }
    }
```

Example: classic CLI

```
A:node-2>config>service# info
```

```
------------------------------------------------
    oper-group "VAwP1" create
    oper-group "VAwP2" create
    exit
    vprn 1 customer 1 create
        shutdown
        interface to-lan
            vrrp 1 create
                oper-group "VAwP1"
            exit
         exit
        interface to-lan2
            vrrp 1 create
                oper-group "VAwP2"
            exit
         exit
         pim
            interface to-lan
                monitor-oper-group "VAwP1" family ipv4 add 90
                monitor-oper-group "VAwP1" family ipv6 add 90
            exit
            interface to-lan2
                monitor-oper-group "VAwP2" family ipv4 add 90
                monitor-oper-group "VAwP2" family ipv6 set 90
            exit
      exit
exit
```

5.2 IPv6 PIM models

IPv6 multicast enables multicast applications over native IPv6 networks. There are two service models: Any Source Multicast (ASM) and Source Specific Multicast (SSM) which includes PIM-SSM and MLD (see MLD overview). SSM does not require source discovery and only supports single source for a specific multicast stream. As a result, SSM is easier to operate in a large scale deployment that uses the one-to-many service model.

5.2.1 PIM-SSM

The Source Specific Multicast (SSM) service model defines a channel identified by an (S,G) pair, where S is a source address and G is an SSM destination address. In contrast to the ASM model, SSM only provides network-layer support for one-to-many delivery.

The SSM service model attempts to alleviate the following deployment problems that ASM has presented:

- **address allocation**

  SSM defines channels on a per-source basis. For example, the channel (S1,G) is distinct from the channel (S2,G), where S1 and S2 are source addresses, and G is an SSM destination address. This averts the problem of global allocation of SSM destination addresses and makes each source independently responsible for resolving address collisions for the various channels it creates.

- **access control**

  SSM provides an efficient solution to the access control problem. When a receiver subscribes to an (S,G) channel, it receives data sent only by the source S. In contrast, any host can transmit to an ASM host group. At the same time, when a sender picks an (S,G) channel to transmit on, it is automatically

ensured that no other sender transmits on the same channel (except in the case of malicious acts such as address spoofing). This makes it harder to spam an SSM channel than an ASM multicast group.

- **handling of well-known sources**

  SSM requires only source-based forwarding trees, eliminating the need for a shared tree infrastructure. In terms of the IGMP, PIM-SM, MSDP, MBGP protocol suite, this implies that neither the RP-based shared tree infrastructure of PIM-SM nor the MSDP protocol is required. Thus, the complexity of the multicast routing infrastructure for SSM is low, making it viable for immediate deployment. MBGP is still required for distribution of multicast reachability information.

- anticipating that point-to-multipoint applications such as Internet TV will be significant in the future, the SSM model is better suited for such applications.

5.2.1.1 System PIM SSM scaling

PIM SSM scaling can be increased to 256k (S,G)s using the **pim-ssm-scaling** command. This command enables (S,G) scaling for PIM SSM in the global routing table only. The current scaling limitation of (S,G)s per complex (FP) still exist. However, the 256K (S,G)s can be configured over multiple complex to achieve this higher scaling.

When PIM SSM scaling is enabled, the following multicast features are disabled:

- DM
- MoFRR
- JP policy
- SSM groups
- (S,G) programming is a maximum of 32000 per complex
- InBand features (BIER and MLDP)
- Extranet
- ASM

This feature is only supported on CPM5s.

When the **pim-ssm-scaling** command is enabled and there is a mix of FP3, FP4, and FP5 cards in the system, Nokia recommends that you configure the following command with the **dynamic** option to ensure the system dynamically chooses the lowest denominator throughput card as multicast-plane throughput.

- **MD-CLI**

```
configure multicast-management chassis-level per-mcast-plane-capacity total-capacity
 dynamic
```

- **classic CLI**

```
configure mcast-management chassis-level per-mcast-plane-capacity total-capacity dynamic
```

Note: When PIM SSM scaling is enabled with IMPM, and different generations of FP are provisioned in the system, Nokia recommends that the multicast-management chassis per-plane total capacity is left at its default value of **dynamic**.

To achieve fast failover when PIM SSM scaling is enabled, the default MCID is used which results in the multicast traffic being sent to all line cards and silently discarded where there is no receiver for that

traffic. Consequently, the maximum achievable plane capacity for this traffic is constrained to that of the lowest performance FP. When the maximum link capacity from the fabric to the lowest-performance FP is reached, the link to that FP is overloaded causing the fabric to back-pressure the ingress and resulting in packet loss for all FPs. By using the default MCID, this capacity constraint is independent of whether the lowest-performance FP has a receiver on it or not.

If the multicast management chassis per-plane total capacity is configured to an explicit value which is larger than that supported by the lowest-performance FP, IMPM believes there is more plane capacity available than there really is and the result is (S,G) packet loss instead of blackholing.

By setting the multicast management chassis per-plane total capacity to **dynamic**, the system automatically sets the switch fabric multicast plane capacity to the minimum value supported by the fabric and all line cards in the system. IMPM then has the correct view of the available plane capacity and correctly blackholes (S,G)s when insufficient plane capacity is available. The total maximum multicast capacity is still constrained by the lowest-performance FP.

5.2.2 PIM ASM

IPv6 PIM ASM is supported. All PIM ASM related functions such as bootstrap router, RP, and so on, support both IPv4 and IPv6 address-families. Use the following command to configure IPv6.

```
configure router pim ipv6
```

5.2.3 Embedded RP

The detailed protocol specification is defined in RFC 3956, *Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address*. This RFC describes a multicast address allocation policy in which the address of the RP is encoded in the IPv6 multicast group address, and specifies a PIM-SM group-to-RP mapping to use the encoding, leveraging, and extending unicast-prefix-based addressing. This mechanism not only provides a simple solution for IPv6 inter-domain ASM but can be used as a simple solution for IPv6 intra-domain ASM with scoped multicast addresses as well. It can also be used as an automatic RP discovery mechanism in those deployment scenarios that would have previously used the Bootstrap Router protocol (BSR).

5.3 Configurable source IP address for PIM register messages

When PIM messages are transmitted over IGP shortcuts, their source IP addresses are selected by choosing the smallest IP address from available interfaces on the node. This can be undesirable because of security measures within the network, such as ACLs, that cause packets to drop. To prevent the messages from being dropped, SR OS supports configuring the source IP address of the register messages to any IP address, regardless of whether it resides on the node.

Use the following commands to configure the source IP address for register messages:

- **MD-CLI**

```
configure router pim ipv4 source-address register-message
configure router pim ipv6 source-address register-message
configure service vprn pim ipv4 source-address register-message
configure service vprn pim ipv6 source-address register-message
```

- **classic CLI**

```
configure router pim source-address register-message
configure service vprn pim source-address register-message
```

5.4 Configuring PIM with CLI

This section provides information to configure PIM using the CLI.

5.4.1 PIM configuration overview

The PIM protocol is not operational until at least one interface is specified for it, at which time the interface is enabled for PIM and is called a PIM interface. When enabled, a PIM interface can be configured with PIM parameters in addition to the standard parameters for the interface when it is created. When PIM is operational, data is forwarded to network segments with active host receivers that have explicitly requested the multicast group.

Note: Before an IP interface can be specified in the PIM context, it must be created using either the **config**>**router**>**interface** or **config**>**service**>**ies**>**interface** command.

5.4.2 Basic PIM configuration

Perform the following basic PIM configuration tasks:

1. Enable PIM (required)
2. Add interfaces so the protocol establishes adjacencies with the neighboring routers (required)
3. Configure a way to calculate group-to-RP mapping (required) by either:
   - static group-to-RP mapping
   - enabling Candidate RP/Bootstrap mechanism on some routers
4. Enable unicast routing protocols to learn routes toward the RP/source for reverse path forwarding (required)
5. Add SSM ranges (optional)
6. Enable Candidate BSR (optional)
7. Enable Candidate RP (optional)
8. Change hello interval (optional)
9. Configure route policies (bootstrap-export, bootstrap-import, import join and register)

5.4.3 PIM configuration

5.4.3.1 Configuring and enabling PIM

When configuring PIM, make sure to enable PIM on all interfaces for the routing instance, otherwise multicast routing errors can occur.

Use the commands in the following context to configure PIM.

```
configure router pim
```

The following example displays a basic configuration with PIM enabled.

Example: MD-CLI

```
[ex:/configure router "Base" pim]
A:admin@node-2# info
    admin-state enable
    apply-to none
    rp {
        ipv4 {
            bsr-candidate {
                admin-state disable
                address 10.10.10.2
                priority 0
                hash-mask-len 30
            }
            rp-candidate {
                admin-state disable
                holdtime 150
                priority 192
                address 10.10.10.1
            }
            static {
                address 10.10.10.10 {
                }
                address 198.51.100.254 {
                    group-prefix 239.24.24.24/32 {
                }
            }
        }
    }
```

Example: classic CLI

```
A:node-2>config>router# info
#------------------------------------------
echo "PIM Configuration"
#------------------------------------------
        pim
            apply-to none
            rp
                static
                    address 198.51.100.254
                        group-prefix 239.24.24.24/32
                    exit
                    address 10.10.10.10
                    exit
```

```
                exit
                bsr-candidate
                    shutdown
                    address 10.10.10.2
                    priority 0
                    hash-mask-len 30
                exit
                rp-candidate
                     shutdown
                     address 10.10.10.1
                     holdtime 150
                     priority 192
                exit
            exit
            no shutdown
        exit
----------------------------------------
```

5.4.3.2 Configuring PIM interfaces

You can reference router interfaces in the PIM configuration. You must create the interfaces first in the router context. Use the commands in the following context to configure and enable PIM router interfaces.

```
configure router pim interface
```

The following example shows a PIM configuration with basic interfaces configured.

Example: MD-CLI

```
[ex:/configure router "base" pim]
A:admin@node-2# info
    admin-state enable
    apply-to all
    interface "lax-sjc" {
        admin-state enable
    }
    interface "lax-vls" {
        admin-state enable
    }
    interface "pl-ix" {
        admin-state enable
    }
    interface "system" {
        admin-state enable
    }
    rp {
        ipv4 {
            bsr-candidate {
                admin-state enable
                address 10.10.10.10
            }
            rp-candidate {
                admin-state enable
                address 10.10.10.1
            }
            static {
                address 10.10.10.1 {
                }
                address 198.51.100.254 {
                    group-prefix 239.24.24.24/32 { }
```

```
                }
            }
        }
    }
```

Example: classic CLI

```
A:node-2>config>router>pim# info
----------------------------------------------
            interface "system"
                no shutdown
            exit
            interface "lax-sjc"
                no shutdown
            exit
            interface "lax-vls"
                no shutdown
            exit
            interface "pl-ix"
                no shutdown
            exit
            apply-to all
            rp
                static
                    address 10.10.10.1
                    exit
                    address 198.51.100.254
                        group-prefix 239.24.24.24/32
                    exit
                exit
                bsr-candidate
                    address 10.10.10.10
                    no shutdown
                exit
                rp-candidate
                    address 10.10.10.1
                    no shutdown
                exit
            exit
            no shutdown
----------------------------------------------
```

5.4.3.3 Configuring PIM join and register policies

Join policies are used in Protocol Independent Multicast (PIM) configurations to prevent the transportation of multicast traffic across a network and the dropping of packets at a scope at the edge of the network. PIM Join filters reduce the potential for denial of service (DoS) attacks and PIM state explosion—large numbers of Joins forwarded to each router on the RPT, resulting in memory consumption. See the Importing PIM Join/Register Policies section of the Multicast Routing Guide for more information.

(*,G) or (S,G) is the information used to forward unicast or multicast packets. The following options can be configured:

- **group-address**

  This matches the group address policy in join/prune messages group-address "group-address-policy”.

- **source-address**

  This is the source-address (192.168.0.1) that matches the source address in join/prune messages.

- **interface**

  This matches any join message received on the specified interface, for example, interface port 1/1/1
- **neighbor**

  This matches any join message received from the specified neighbor; for example, neighbor 1.1.1.1

Use commands in the following context to configure policy options:

- **MD-CLI**

```
configure policy-options
```

- **classic CLI**

```
configure router policy-options
```

The following configuration example does not allow join messages for the specified group address prefix list and source 192.168.0.1 but allows other join messages.

Example: MD-CLI

```
[ex:/configure policy-options]
A:admin@cses-V208# info
    prefix-list "prefix-list-1" {
        prefix 192.0.2.0/24 type exact {
        }
    }
    policy-statement "Foo" {
        entry 10 {
            from {
                group-address "prefix-list-1"
                source-address {
                    ip-address 192.168.0.1
                }
            }
        }
    }
```

Example: classic CLI

```
A:node-2>config>router>policy-options# info
----------------------------------------------
            prefix-list "prefix-list-1"
                prefix 192.0.2.0/24 exact
            exit
            policy-statement "Foo"
                entry 10
                    from
                        group-address "prefix-list-1"
                        source-address 192.168.0.1
                    exit
                exit
            exit
----------------------------------------------
```

5.4.3.4 Importing PIM join and register policies

An import mechanism is provided to control the (*,G) and (S,G) states that are created on the router.

Note: In the import policy, if an action is not specified in the entry then the default-action takes precedence. If no entry matches then the default-action also takes precedence. If no default-action is specified, then the default default-action is executed.

Use the following commands to configure PIM join or register import policies.

```
configure router pim import join-policy
configure router pim import register-policy
```

The following example shows a PIM configuration with an imported policy applied. The policy would not allow join messages for group 229.50.50.208/32 and source 192.168.0.0/16 but would allow join messages for 192.168.0.0/16, 229.50.50.208 (see the "Configuring Route Policy Components" section of the *7705 SAR Gen 2 Unicast Routing Protocols Guide*).

Example: MD-CLI

```
[ex:/configure router "base" pim]
A:admin@node-2# info
    ...
    apply-to-all true
    import join-policy "foo"
    interface "lax-sjc" {
        admin-state enable
    }
    interface "lax-vls" {
        admin-state enable
    }
    interface "pl-ix" {
        admin-state enable
    }
    interface "system" {
        admin-state enable
    }
    rp {
        ipv4 {
            bsr-candidate {
                admin-state enable
                address 10.10.10.10
            }
            rp-candidate {
                admin-state enable
                address 10.10.10.1
            }
            static {
                address 10.10.10.1 {
                }
                address 198.51.100.254 {
                    group-prefix 239.24.24.24/32 { }
                }
            }
        }
    }
    ...
```

Example: classic CLI

```
A:node-2>config>router>pim# info
----------------------------------------------
            ...
            import join-policy "foo"
```

```
            interface "system"
            exit
            interface "lax-sjc"
            exit
            interface "lax-vls"
            exit
            interface "pl-ix"
            exit
            apply-to all
            rp
                static
                    address 10.10.10.1
                    exit
                    address 198.51.100.254
                        group-prefix 239.24.24.24/32
                    exit
                exit
                bsr-candidate
                    address 10.10.10.10
                    no shutdown
                exit
                rp-candidate
                    address 10.10.10.1
                    no shutdown
                exit
            exit
            ...
-------------------------------------------
```

5.4.3.5 Configuring bootstrap message import and export policies

Bootstrap import and export policies are used to control the flow of bootstrap messages to and from the RP.

The following configuration example specifies that no BSR messages are received or sent out of interface port 1/1/1.

Example: Configuration of import and export policy statements (MD-CLI)

```
[ex:/configure policy-options]
A:admin@node-2# info
    ...
    prefix-list "pim-policy-1" {
        prefix 10.0.0.0/16 longer
        prefix 10.10.186.0/24 longer
    }
    prefix-list "pim-policy-2"  {
        prefix 10.1.0.0/16 longer
    }
    policy-statement "pim-export-policy" {
        entry 10 {
            to {
                prefix-list "pim-policy-1" "pim-policy-2"
            }
            action {
                action-type reject
            }
        }
    }
    policy-statement "pim-import-policy" {
        entry 10 {
```

```
            from {
                interface ["port1"]
            }
            action {
                action-type reject
            }
        }
    }
    ...
```

Example: Configuration of import and export policy statements (classic CLI)

```
A:node-2>config>router>policy-options# info
----------------------------------------------
        ...
        prefix-list "pim-policy-1"
            prefix 10.0.0.0/16 longer
            prefix 10.10.186.0/24 longer
        exit
        prefix-list "pim-policy-2"  {
            prefix 10.1.0.0/16 longer
        exit
        policy-statement "pim-import-policy"
            entry 10
                from
                    interface "port1"
                exit
                action drop
                exit
            exit
        exit
        policy-statement "pim-export-policy"
            entry 10
                to
                    prefix-list "pim-policy-1" "pim-policy-2"
                exit
                action accept
            exit
        exit
        ...
```

Example: PIM configuration with import and export policies (MD-CLI)

```
[ex:/configure router "Base" pim]
A:node-2# info
    admin-state enable
    apply-to all
    interface "lax-sjc" {
    }
    interface "lax-vls" {
    }
    interface "pl-ix" {
    }
    interface "system" {
    }
    rp {
        bootstrap {
            import ["pim-import"]
        }
        bootstrap {
            export ["pim-export"]
        }
```

```
        ipv4 {
            bsr-candidate {
                admin-state disable
                priority 0
                address 10.10.10.10
                hash-mask-len 30
            }
            rp-candidate {
                admin-state enable
                address 10.10.10.1
            }
            static {
                address 10.10.10.1 {
                }
                address 198.51.100.254 {
                    group-prefix 239.24.24.24/32 { }
                }
            }
        }
    }
```

Example: PIM configuration with import and export policies (classic CLI)

```
A:node-2>config>router>pim# info
----------------------------------------------
            interface "system"
            exit
            interface "lax-sjc"
            exit
            interface "lax-vls"
            exit
            interface "pl-ix"
            exit
            apply-to all
            rp
                bootstrap-import "pim-import"
                bootstrap-export "pim-export"
                static
                    address 10.10.10.1
                    exit
                    address 198.51.100.254
                        group-prefix 239.24.24.24/32
                    exit
                exit
                bsr-candidate
                    shutdown
                    address 10.10.10.10
                exit
                rp-candidate
                    address 10.10.10.1
                    no shutdown
                exit
            exit
            no shutdown
----------------------------------------------
```

5.4.4 Disabling PIM

Use the following commands to disable PIM:

- **MD-CLI**

```
configure router pim admin-state disable
```

- **classic CLI**

```
configure router pim shutdown
```

6 Troubleshooting tools

This chapter provides information about troubleshooting tools.

6.1 Mtrace

To help assess problems in the distribution of IP multicast traffic, the **mtrace** feature uses a traceroute feature implemented in multicast routers that is accessed via an extension to the IGMP protocol. The **mtrace** feature is used to print the path from the source to a receiver; it does this by passing a trace query hop-by-hop along the reverse path from the receiver to the source. At each hop, information such as the hop address, routing error conditions, and packet statistics should be gathered and returned to the requester.

Data added by each hop includes:

- query arrival time
- incoming interface
- outgoing interface
- previous hop router address
- input packet count
- output packet count
- total packets for this source/group
- routing protocol
- Time To Live (TTL) threshold
- forwarding/error code

The information enables the network administrator to determine:

- where multicast flows stop
- the flow of the multicast stream

When the trace response packet reaches the FHR (the router that is directly connected to the net of the source), that router sends the completed response to the response destination (receiver) address specified in the trace query.

If some multicast router along the path does not implement the multicast traceroute feature, or if there is some outage, no response is returned. To solve this problem, the trace query includes a maximum hop count field to limit the number of hops traced before the response is returned. This allows a partial path to be traced.

The reports inserted by each router contain not only the address of the hop, but also the TTL required to forward, and some flags to indicate routing errors, plus counts of the total number of packets on the incoming and outgoing interfaces, and those forwarded for the specified group. Taking differences in these counts for two traces separated in time, and comparing the output packet counts from one hop with the input packet counts of the next-hop, allows the calculation of packet rate and packet loss statistics for each hop, to isolate congestion problems.

6.1.1 Finding the last hop router

The trace query must be sent to the multicast router which is the last hop on the path from the source to the receiver. If the receiver is on the local subnet (as determined using the subnet mask), then the default method is to multicast the trace query to all-routers.mcast.net (224.0.0.2) with a TTL of 1. Otherwise, the trace query is multicast to the group address because the last hop router is a member of that group if the receiver is. Therefore, it is necessary to specify a group that the intended receiver has joined. This multicast is sent with a default TTL of 64, which may not be sufficient for all cases.

When tracing from a multihomed host or router, the default receiver address may not be the wanted interface for the path from the source. In that case, the wanted interface should be specified explicitly as the receiver.

6.1.2 Directing the response

By default, **mtrace** first attempts to trace the full reverse path, unless the number of hops to trace is explicitly set with the hop option. If there is no response within a 3 second timeout interval, an asterisk (*) is printed, and the probing switches to hop-by-hop mode. Trace queries are issued starting with a maximum hop count of one and increasing by one until the full path is traced or no response is received. At each hop, multiple probes are sent. The first attempt is made with the unicast address of the host running **mtrace** as the destination for the response. As the unicast route may be blocked, the remainder of attempts request that the response be multicast to mtrace.mcast.net (224.0.1.32) with the TTL set to 32 more than what is needed to pass the thresholds seen so far along the path to the receiver. For the last attempts, the TTL is increased by another 32.

Alternatively, the TTL may be set explicitly with the TTL option.

For each attempt, if no response is received within the timeout, an asterisk (*) is printed. After the specified number of attempts have failed, **mtrace** tries to query the next-hop router with a DVMRP_ASK_NEIGHBORS2 request (as used by the **mrinfo** program) to determine the router type.

The output of **mtrace** is a short listing of the hops in the order they are queried, that is, in the reverse of the order from the source to the receiver. For each hop, a line is printed showing:

- hop number (counted negatively to indicate that this is the reverse path)
- multicast protocol
- threshold required to forward data (to the previous hop in the listing as indicated by the up-arrow character)
- cumulative delay for the query to reach that hop (valid only if the clocks are synchronized)

The response ends with a line showing the round-trip time, which measures the interval from when the query was issued until the response was received, both derived from the local system clock.

Mtrace packets use special IGMP packets with IGMP type codes of 0x1E and 0x1F.

6.2 Mstat

The **mstat** command adds the capability to show the multicast path in a limited graphic display and provide drops, duplicates, TTLs and delays at each node. This information is useful to the network user because it identifies nodes with high drop and duplicate counts. Duplicate counts are shown as negative drops.

The output of **mstat** provides a limited pictorial view of the path in the forward direction with data flow indicated by arrows pointing downward and the query path indicated by arrows pointing upward. For each hop, both the entry and exit addresses of the router are shown if different, along with the initial ttl required on the packet to be forwarded at this hop and the propagation delay across the hop assuming that the routers at both ends have synchronized clocks. The output consists of two columns, one for the overall multicast packet rate that does not contain lost/sent packets and a column for the (S,G)-specific case. The S,G statistics do not contain lost/sent packets.

6.3 Mrinfo

Mrinfo is a mechanism based on the **ask_neighbors**>**igmp** command to display the configuration information from the target multicast router. The type of information displayed includes the multicast capabilities of the router, code version, metrics, ttl-thresholds, protocols, and status. This information, for instance, can be used by network operators to verify whether bidirectional adjacencies exist. When the specified multicast router responds, the configuration is displayed.

7 Standards and protocol support

Note:
The information provided in this chapter is subject to change without notice and may not apply to all platforms.

Nokia assumes no responsibility for inaccuracies.

7.1 Bidirectional Forwarding Detection (BFD)

RFC 5880, *Bidirectional Forwarding Detection (BFD)*

RFC 5881, *Bidirectional Forwarding Detection (BFD) IPv4 and IPv6 (Single Hop)*

RFC 5882, *Generic Application of Bidirectional Forwarding Detection (BFD)*

7.2 Border Gateway Protocol (BGP)

draft-hares-idr-update-attrib-low-bits-fix-01, *Update Attribute Flag Low Bits Clarification*

draft-ietf-idr-add-paths-guidelines-08, *Best Practices for Advertisement of Multiple Paths in IBGP*

draft-ietf-idr-best-external-03, *Advertisement of the best external route in BGP*

draft-ietf-idr-bgp-gr-notification-01, *Notification Message support for BGP Graceful Restart*

draft-ietf-idr-bgp-optimal-route-reflection-10, *BGP Optimal Route Reflection (BGP-ORR)*

draft-ietf-idr-error-handling-03, *Revised Error Handling for BGP UPDATE Messages*

draft-ietf-idr-link-bandwidth-03, *BGP Link Bandwidth Extended Community*

RFC 1772, *Application of the Border Gateway Protocol in the Internet*

RFC 1997, *BGP Communities Attribute*

RFC 2385, *Protection of BGP Sessions via the TCP MD5 Signature Option*

RFC 2439, *BGP Route Flap Damping*

RFC 2545, *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*

RFC 2858, *Multiprotocol Extensions for BGP-4*

RFC 2918, *Route Refresh Capability for BGP-4*

RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*

RFC 4360, *BGP Extended Communities Attribute*

RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 4456, *BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)*

RFC 4486, *Subcodes for BGP Cease Notification Message*

RFC 4659, *BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN*

RFC 4684, *Constrained Route Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)*

RFC 4724, *Graceful Restart Mechanism for BGP* – helper mode

RFC 4760, *Multiprotocol Extensions for BGP-4*

RFC 4798, *Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)*

RFC 5004, *Avoid BGP Best Path Transitions from One External to Another*

RFC 5065, *Autonomous System Confederations for BGP*

RFC 5291, *Outbound Route Filtering Capability for BGP-4*

RFC 5396, *Textual Representation of Autonomous System (AS) Numbers* – asplain

RFC 5492, *Capabilities Advertisement with BGP-4*

RFC 5668, *4-Octet AS Specific BGP Extended Community*

RFC 6286, *Autonomous-System-Wide Unique BGP Identifier for BGP-4*

RFC 6368, *Internal BGP as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 6793, *BGP Support for Four-Octet Autonomous System (AS) Number Space*

RFC 6810, *The Resource Public Key Infrastructure (RPKI) to Router Protocol*

RFC 6811, *Prefix Origin Validation*

RFC 6996, *Autonomous System (AS) Reservation for Private Use*

RFC 7311, *The Accumulated IGP Metric Attribute for BGP*

RFC 7606, *Revised Error Handling for BGP UPDATE Messages*

RFC 7607, *Codification of AS 0 Processing*

RFC 7752, *North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP*

RFC 7911, *Advertisement of Multiple Paths in BGP*

RFC 7999, *BLACKHOLE Community*

RFC 8092, *BGP Large Communities Attribute*

RFC 8097, *BGP Prefix Origin Validation State Extended Community*

RFC 8212, *Default External BGP (EBGP) Route Propagation Behavior without Policies*

RFC 8277, *Using BGP to Bind MPLS Labels to Address Prefixes*

RFC 9294, *Application-Specific Link Attributes Advertisement Using the Border Gateway Protocol - Link State (BGP LS)*

RFC 9494, *Long-Lived Graceful Restart for BGP*

7.3 Bridging and management

IEEE 802.1AB, *Station and Media Access Control Connectivity Discovery*

IEEE 802.1ad, *Provider Bridges*

IEEE 802.1AX, *Link Aggregation*

IEEE 802.1D, *MAC Bridges*

IEEE 802.1p, *Traffic Class Expediting*

IEEE 802.1Q, *Virtual LANs*

IEEE 802.1s, *Multiple Spanning Trees*

IEEE 802.1w, *Rapid Reconfiguration of Spanning Tree*

7.4 Certificate management

RFC 4210, *Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)*

RFC 4211, *Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)*

RFC 5280, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*

RFC 6712, *Internet X.509 Public Key Infrastructure -- HTTP Transfer for the Certificate Management Protocol (CMP)*

RFC 7030, *Enrollment over Secure Transport*

RFC 7468, *Textual Encodings of PKIX, PKCS, and CMS Structures*

7.5 Ethernet

IEEE 802.3x, *Ethernet Flow Control*

7.6 Ethernet VPN (EVPN)

draft-ietf-bess-evpn-ipvpn-interworking-15, *EVPN Interworking with IPVPN*

draft-ietf-bess-evpn-l3mh-proto-00, *EVPN Multi-Homing support for L3 services*

draft-rbickhart-evpn-ip-mac-proxy-adv-04, *Proxy MAC-IP Advertisement in EVPN*

RFC 7432, *BGP MPLS-Based Ethernet VPN*

RFC 8214, *Virtual Private Wire Service Support in Ethernet VPN*

RFC 8365, *A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)*

RFC 8560, *Seamless Integration of Ethernet VPN (EVPN) with Virtual Private LAN Service (VPLS) and Their Provider Backbone Bridge (PBB) Equivalents*

RFC 9047, *Propagation of ARP/ND Flags in an Ethernet Virtual Private Network (EVPN)*

RFC 9135, *Integrated Routing and Bridging in Ethernet VPN (EVPN)*

RFC 9136, *IP Prefix Advertisement in Ethernet VPN (EVPN)*

RFC 9161, *Operational Aspects of Proxy ARP/ND in Ethernet Virtual Private Networks*

RFC 9251, *Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Proxies for Ethernet VPN (EVPN)*

7.7 gRPC Remote Procedure Calls (gRPC)

cert.proto version 0.1.0, *gNOI Certificate Management Service*

file.proto version 0.1.0, *gNOI File Service*

gnmi.proto version 0.8.0, *gNMI Service Specification*

gnmi_ext.proto, *gNMI Commit Confirmed Extension*

gnmi_ext.proto, *gNMI Config Subscription Extension*

gnmi_ext.proto, *gNMI Depth Extension*

system.proto version 1.0.0, *gNOI System Service*

tunnel.proto version 0.2, *gRPC Tunnel Service*

PROTOCOL-HTTP2, *gRPC over HTTP2*

7.8 Intermediate System to Intermediate System (IS-IS)

draft-ietf-isis-mi-02, *IS-IS Multi-Instance*

draft-ietf-lsr-igp-ureach-prefix-announce-01, *IGP Unreachable Prefix Announcement* – without U-Flag and UP-Flag

draft-kaplan-isis-ext-eth-02, *Extended Ethernet Frame Size Support*

ISO/IEC 10589:2002 Second Edition, *Intermediate system to Intermediate system intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)*

RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*

RFC 2973, *IS-IS Mesh Groups*

RFC 3359, *Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System*

RFC 3719, *Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)*

RFC 3787, *Recommendations for Interoperable IP Networks using Intermediate System to Intermediate System (IS-IS)*

RFC 5120, *M-ISIS: Multi Topology (MT) Routing in IS-IS*

RFC 5130, *A Policy Control Mechanism in IS-IS Using Administrative Tags*

RFC 5301, *Dynamic Hostname Exchange Mechanism for IS-IS*

RFC 5302, *Domain-wide Prefix Distribution with Two-Level IS-IS*

RFC 5303, *Three-Way Handshake for IS-IS Point-to-Point Adjacencies*

RFC 5304, *IS-IS Cryptographic Authentication*

RFC 5305, *IS-IS Extensions for Traffic Engineering TE*

RFC 5306, *Restart Signaling for IS-IS* – helper mode

RFC 5308, *Routing IPv6 with IS-IS*

RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*

RFC 5310, *IS-IS Generic Cryptographic Authentication*

RFC 6213, *IS-IS BFD-Enabled TLV*

RFC 6232, *Purge Originator Identification TLV for IS-IS*

RFC 6233, *IS-IS Registry Extension for Purges*

RFC 7775, *IS-IS Route Preference for Extended IP and IPv6 Reachability*

RFC 7794, *IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability* – sections 2.1 and 2.3

RFC 7981, *IS-IS Extensions for Advertising Router Information*

RFC 7987, *IS-IS Minimum Remaining Lifetime*

RFC 8202, *IS-IS Multi-Instance* – single topology

RFC 8570, *IS-IS Traffic Engineering (TE) Metric Extensions* – Min/Max Unidirectional Link Delay metric for flex-algo, RSVP, SR-TE

RFC 8919, *IS-IS Application-Specific Link Attributes*

RFC 9885, *Multi-Part TLVs in IS-IS*

7.9 Internet Protocol (IP) general

RFC 768, *User Datagram Protocol*

RFC 793, *Transmission Control Protocol*

RFC 854, *Telnet Protocol Specifications*

RFC 1350, *The TFTP Protocol (revision 2)*

RFC 2784, *Generic Routing Encapsulation (GRE)*

RFC 3164, *The BSD syslog Protocol*

RFC 4250, *The Secure Shell (SSH) Protocol Assigned Numbers*

RFC 4251, *The Secure Shell (SSH) Protocol Architecture*

RFC 4252, *The Secure Shell (SSH) Authentication Protocol* – publickey, password

RFC 4253, *The Secure Shell (SSH) Transport Layer Protocol*

RFC 4254, *The Secure Shell (SSH) Connection Protocol*

RFC 4511, *Lightweight Directory Access Protocol (LDAP): The Protocol*

RFC 4513, *Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms* – TLS

RFC 4632, *Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan*

RFC 5082, *The Generalized TTL Security Mechanism (GTSM)*

RFC 5246, *The Transport Layer Security (TLS) Protocol Version 1.2* – TLS client, RSA public key

RFC 5289, *TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)*

RFC 5425, *Transport Layer Security (TLS) Transport Mapping for Syslog* – RFC 3164 with TLS

RFC 5656, *Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer* – ECDSA

RFC 5925, *The TCP Authentication Option*

RFC 5926, *Cryptographic Algorithms for the TCP Authentication Option (TCP-AO)*

RFC 6398, *IP Router Alert Considerations and Usage* – MLD

RFC 6528, *Defending against Sequence Number Attacks*

RFC 8446, *The Transport Layer Security (TLS) Protocol Version 1.3*

RFC 8907, *The Terminal Access Controller Access-Control System Plus (TACACS+) Protocol*

7.10 Internet Protocol (IP) multicast

RFC 1112, *Host Extensions for IP Multicasting*

RFC 2236, *Internet Group Management Protocol, Version 2*

RFC 2365, *Administratively Scoped IP Multicast*

RFC 2375, *IPv6 Multicast Address Assignments*

RFC 2710, *Multicast Listener Discovery (MLD) for IPv6*

RFC 3376, *Internet Group Management Protocol, Version 3*

RFC 3446, *Anycast Rendevous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)*

RFC 3590, *Source Address Selection for the Multicast Listener Discovery (MLD) Protocol*

RFC 3618, *Multicast Source Discovery Protocol (MSDP)*

RFC 3810, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*

RFC 4541, *Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches*

RFC 4604, *Using Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Protocol Version 2 (MLDv2) for Source-Specific Multicast*

RFC 4610, *Anycast-RP Using Protocol Independent Multicast (PIM)*

RFC 4611, *Multicast Source Discovery Protocol (MSDP) Deployment Scenarios*

RFC 5059, *Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)*

RFC 5186, *Internet Group Management Protocol Version 3 (IGMPv3) / Multicast Listener Discovery Version 2 (MLDv2) and Multicast Routing Protocol Interaction*

RFC 5384, *The Protocol Independent Multicast (PIM) Join Attribute Format*

RFC 7761, *Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*

RFC 8487, *Mtrace Version 2: Traceroute Facility for IP Multicast*

7.11 Internet Protocol (IP) version 4

RFC 791, *Internet Protocol*

RFC 792, *Internet Control Message Protocol*

RFC 826, *An Ethernet Address Resolution Protocol*

RFC 1034, *Domain Names - Concepts and Facilities*

RFC 1035, *Domain Names - Implementation and Specification*

RFC 1191, *Path MTU Discovery* – router specification

RFC 1519, *Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*

RFC 1812, *Requirements for IPv4 Routers*

RFC 1918, *Address Allocation for Private Internets*

RFC 2131, *Dynamic Host Configuration Protocol*; Relay only

RFC 2132, *DHCP Options and BOOTP Vendor Extensions* – DHCP

RFC 2401, *Security Architecture for Internet Protocol*

RFC 3021, *Using 31-Bit Prefixes on IPv4 Point-to-Point Links*

RFC 3046, *DHCP Relay Agent Information Option (Option 82)*

RFC 3768, *Virtual Router Redundancy Protocol (VRRP)*

RFC 4884, *Extended ICMP to Support Multi-Part Messages* – ICMPv4 and ICMPv6 Time Exceeded

7.12 Internet Protocol (IP) version 6

RFC 2464, *Transmission of IPv6 Packets over Ethernet Networks*

RFC 2529, *Transmission of IPv6 over IPv4 Domains without Explicit Tunnels*

RFC 3122, *Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification*

RFC 3315, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*

RFC 3587, *IPv6 Global Unicast Address Format*

RFC 3596, *DNS Extensions to Support IP version 6*

RFC 3633, *IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6*

RFC 3736, *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*

RFC 3971, *SEcure Neighbor Discovery (SEND)*

RFC 4007, *IPv6 Scoped Address Architecture*

RFC 4191, *Default Router Preferences and More-Specific Routes* – Default Router Preference

RFC 4193, *Unique Local IPv6 Unicast Addresses*

RFC 4291, *Internet Protocol Version 6 (IPv6) Addressing Architecture*

RFC 4443, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*

RFC 4861, *Neighbor Discovery for IP version 6 (IPv6)*

RFC 5095, *Deprecation of Type 0 Routing Headers in IPv6*

RFC 5722, *Handling of Overlapping IPv6 Fragments*

RFC 5798, *Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6* – IPv6

RFC 5952, *A Recommendation for IPv6 Address Text Representation*

RFC 6164, *Using 127-Bit IPv6 Prefixes on Inter-Router Links*

RFC 8021, *Generation of IPv6 Atomic Fragments Considered Harmful*

RFC 8200, *Internet Protocol, Version 6 (IPv6) Specification*

7.13 Internet Protocol Security (IPsec)

draft-ietf-ipsec-isakmp-mode-cfg-05, *The ISAKMP Configuration Method*

draft-ietf-ipsec-isakmp-xauth-06, *Extended Authentication within ISAKMP/Oakley (XAUTH)*

RFC 2401, *Security Architecture for the Internet Protocol*

RFC 2403, *The Use of HMAC-MD5-96 within ESP and AH*

RFC 2404, *The Use of HMAC-SHA-1-96 within ESP and AH*

RFC 2405, *The ESP DES-CBC Cipher Algorithm With Explicit IV*

RFC 2406, *IP Encapsulating Security Payload (ESP)*

RFC 2407, *IPsec Domain of Interpretation for ISAKMP (IPsec DoI)*

RFC 2408, *Internet Security Association and Key Management Protocol (ISAKMP)*

RFC 2409, *The Internet Key Exchange (IKE)*

RFC 2410, *The NULL Encryption Algorithm and Its Use With IPsec*

RFC 2560, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*

RFC 3526, *More Modular Exponential (MODP) Diffie-Hellman group for Internet Key Exchange (IKE)*

RFC 3566, *The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec*

RFC 3602, *The AES-CBC Cipher Algorithm and Its Use with IPsec*

RFC 3706, *A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers*

RFC 3947, *Negotiation of NAT-Traversal in the IKE*

RFC 3948, *UDP Encapsulation of IPsec ESP Packets*

RFC 4106, *The Use of Galois/Counter Mode (GCM) in IPsec ESP*

RFC 4109, *Algorithms for Internet Key Exchange version 1 (IKEv1)*

RFC 4301, *Security Architecture for the Internet Protocol*

RFC 4303, *IP Encapsulating Security Payload*

RFC 4307, *Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2)*

RFC 4308, *Cryptographic Suites for IPsec*

RFC 4434, *The AES-XCBC-PRF-128 Algorithm for the Internet Key Exchange Protocol (IKE)*

RFC 4543, *The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH*

RFC 4754, *IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA)*

RFC 4835, *Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)*

RFC 4868, *Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec*

RFC 4945, *The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2 and PKIX*

RFC 5019, *The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments*

RFC 5282, *Using Authenticated Encryption Algorithms with the Encrypted Payload of the IKEv2 Protocol*

RFC 5903, *ECP Groups for IKE and IKEv2*

RFC 5996, *Internet Key Exchange Protocol Version 2 (IKEv2)*

RFC 5998, *An Extension for EAP-Only Authentication in IKEv2*

RFC 6379, *Suite B Cryptographic Suites for IPsec*

RFC 6380, *Suite B Profile for Internet Protocol Security (IPsec)*

RFC 6960, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*

RFC 7296, *Internet Key Exchange Protocol Version 2 (IKEv2)*

RFC 7321, *Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)*

RFC 7383, *Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation*

RFC 7427, *Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)*

RFC 8784, *Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security*

7.14 Label Distribution Protocol (LDP)

draft-pdutta-mpls-ldp-adj-capability-00, *LDP Adjacency Capabilities*

draft-pdutta-mpls-ldp-v2-00, *LDP Version 2*

RFC 3037, *LDP Applicability*

RFC 3478, *Graceful Restart Mechanism for Label Distribution Protocol* – helper mode

RFC 5036, *LDP Specification*

RFC 5283, *LDP Extension for Inter-Area Label Switched Paths (LSPs)*

RFC 5443, *LDP IGP Synchronization*

RFC 5561, *LDP Capabilities*

RFC 5919, *Signaling LDP Label Advertisement Completion*

7.15 Multiprotocol Label Switching (MPLS)

RFC 3031, *Multiprotocol Label Switching Architecture*

RFC 3032, *MPLS Label Stack Encoding*

RFC 3270, *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services* – E-LSP

RFC 3443, *Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks*

RFC 5332, *MPLS Multicast Encapsulations*

RFC 5884, *Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)*

RFC 6424, *Mechanism for Performing Label Switched Path Ping (LSP Ping) over MPLS Tunnels*

RFC 7308, *Extended Administrative Groups in MPLS Traffic Engineering (MPLS-TE)*

RFC 7746, *Label Switched Path (LSP) Self-Ping*

7.16 Network Address Translation (NAT)

RFC 4787, *Network Address Translation (NAT) Behavioral Requirements for Unicast UDP*

RFC 5382, *NAT Behavioral Requirements for TCP*

RFC 5508, *NAT Behavioral Requirements for ICMP*

7.17 Network Configuration Protocol (NETCONF)

RFC 5277, *NETCONF Event Notifications*

RFC 6020, *YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)*

RFC 6022, *YANG Module for NETCONF Monitoring*

RFC 6241, *Network Configuration Protocol (NETCONF)*

RFC 6242, *Using the NETCONF Protocol over Secure Shell (SSH)*

RFC 6243, *With-defaults Capability for NETCONF*

RFC 8071, *NETCONF Call Home and RESTCONF Call Home* – NETCONF

RFC 8342, *Network Management Datastore Architecture (NMDA)* – Startup, Candidate, Running and Intended datastores

RFC 8525, *YANG Library*

RFC 8526, *NETCONF Extensions to Support the Network Management Datastore Architecture* – <get-data> operation

7.18 Media sanitization

NIST Special Publication 800-88 Revision 1, *Guidelines for Media Sanitization* – CF, MMC, SSD, SD, USB

7.19 Open Shortest Path First (OSPF)

RFC 1765, *OSPF Database Overflow*

RFC 2328, *OSPF Version 2*

RFC 3101, *The OSPF Not-So-Stubby Area (NSSA) Option*

RFC 3509, *Alternative Implementations of OSPF Area Border Routers*

RFC 3623, *Graceful OSPF Restart Graceful OSPF Restart* – helper mode

RFC 3630, *Traffic Engineering (TE) Extensions to OSPF Version 2*

RFC 4811, *OSPF Out-of-Band Link State Database (LSDB) Resynchronization* – OSPFv2

RFC 4812, *OSPF Restart Signaling* – OSPFv2

RFC 4576, *Using a Link State Advertisement (LSA) Options Bit to Prevent Looping in BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 4577, *OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 5185, *OSPF Multi-Area Adjacency*

RFC 5243, *OSPF Database Exchange Summary List Optimization*

RFC 5250, *The OSPF Opaque LSA Option*

RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*

RFC 5642, *Dynamic Hostname Exchange Mechanism for OSPF*

RFC 6549, *OSPFv2 Multi-Instance Extensions*

RFC 6987, *OSPF Stub Router Advertisement*

RFC 7471, *OSPF Traffic Engineering (TE) Metric Extensions* – Min/Max Unidirectional Link Delay metric for flex-algo, RSVP, SR-TE

RFC 7684, *OSPFv2 Prefix/Link Attribute Advertisement*

RFC 7770, *Extensions to OSPF for Advertising Optional Router Capabilities*

RFC 8920, *OSPF Application-Specific Link Attributes*

7.20 Path Computation Element Protocol (PCEP)

draft-alvarez-pce-path-profiles-04, *PCE Path Profiles*

draft-ietf-pce-binding-label-sid-15, *Carrying Binding Label/Segment Identifier (SID) in PCE-based Networks.* – MPLS binding SIDs

RFC 5440, *Path Computation Element (PCE) Communication Protocol (PCEP)*

RFC 8231, *Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE*

RFC 8253, *PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)*

RFC 8281, *PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model*

RFC 8408, *Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages*

RFC 8664, *Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing*

7.21 Pseudowire (PW)

draft-ietf-l2vpn-vpws-iw-oam-04, *OAM Procedures for VPWS Interworking*

MFA Forum 12.0.0, *Multiservice Interworking - Ethernet over MPLS*

MFA Forum 13.0.0, *Fault Management for Multiservice Interworking v1.0*

MFA Forum 16.0.0, *Multiservice Interworking - IP over MPLS*

RFC 3916, *Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)*

RFC 3985, *Pseudo Wire Emulation Edge-to-Edge (PWE3)*

RFC 4385, *Pseudo Wire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN*

RFC 4446, *IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)*

RFC 4447, *Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*

RFC 4448, *Encapsulation Methods for Transport of Ethernet over MPLS Networks*

RFC 5085, *Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires*

RFC 5659, *An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge*

RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*

RFC 6073, *Segmented Pseudowire*

RFC 6310, *Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping*

RFC 6391, *Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network*

RFC 6575, *Address Resolution Protocol (ARP) Mediation for IP Interworking of Layer 2 VPNs*

RFC 6718, *Pseudowire Redundancy*

RFC 6829, *Label Switched Path (LSP) Ping for Pseudowire Forwarding Equivalence Classes (FECs) Advertised over IPv6*

RFC 6870, *Pseudowire Preferential Forwarding Status bit*

RFC 7023, *MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking*

RFC 7267, *Dynamic Placement of Multi-Segment Pseudowires*

RFC 7392, *Explicit Path Routing for Dynamic Multi-Segment Pseudowires* – ER-TLV and ER-HOP IPv4 Prefix

RFC 8395, *Extensions to BGP-Signaled Pseudowires to Support Flow-Aware Transport Labels*

7.22 Quality of Service (QoS)

RFC 2430, *A Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)*

RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*

RFC 2597, *Assured Forwarding PHB Group*

RFC 3140, *Per Hop Behavior Identification Codes*

RFC 3246, *An Expedited Forwarding PHB (Per-Hop Behavior)*

7.23 Remote Authentication Dial In User Service (RADIUS)

draft-oscca-cfrg-sm3-02, *The SM3 Cryptographic Hash Function*

RFC 2865, *Remote Authentication Dial In User Service (RADIUS)*

RFC 2866, *RADIUS Accounting*

RFC 3162, *RADIUS and IPv6*

RFC 6613, *RADIUS over TCP* – with TLS

RFC 6614, *Transport Layer Security (TLS) Encryption for RADIUS*

RFC 6929, *Remote Authentication Dial-In User Service (RADIUS) Protocol Extensions*

7.24 Resource Reservation Protocol - Traffic Engineering (RSVP-TE)

RFC 2702, *Requirements for Traffic Engineering over MPLS*

RFC 2747, *RSVP Cryptographic Authentication*

RFC 2961, *RSVP Refresh Overhead Reduction Extensions*

RFC 3097, *RSVP Cryptographic Authentication -- Updated Message Type Value*

RFC 3209, *RSVP-TE: Extensions to RSVP for LSP Tunnels*

RFC 4124, *Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering*

RFC 4125, *Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*

RFC 4127, *Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*

RFC 4561, *Definition of a Record Route Object (RRO) Node-Id Sub-Object*

RFC 5817, *Graceful Shutdown in MPLS and Generalized MPLS Traffic Engineering Networks*

7.25 Routing Information Protocol (RIP)

RFC 1058, *Routing Information Protocol*

RFC 2080, *RIPng for IPv6*

RFC 2082, *RIP-2 MD5 Authentication*

RFC 2453, *RIP Version 2*

7.26 Segment Routing (SR)

RFC 8287, *Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes*

RFC 8426, *Recommendations for RSVP-TE and Segment Routing (SR) Label Switched Path (LSP) Coexistence*

RFC 8476, *Signaling Maximum SID Depth (MSD) Using OSPF* – node MSD

RFC 8491, *Signaling Maximum SID Depth (MSD) Using IS-IS* – node MSD

RFC 8660, *Segment Routing with the MPLS Data Plane*

RFC 8661, *Segment Routing MPLS Interworking with LDP*

RFC 8665, *OSPF Extensions for Segment Routing*

RFC 8667, *IS-IS Extensions for Segment Routing*

RFC 8669, *Segment Routing Prefix Segment Identifier Extensions for BGP*

RFC 9256, *Segment Routing Policy Architecture*

RFC 9350, *IGP Flexible Algorithm*

7.27 Simple Network Management Protocol (SNMP)

draft-blumenthal-aes-usm-04, *The AES Cipher Algorithm in the SNMP's User-based Security Model* – CFB128-AES-192 and CFB128-AES-256

draft-ietf-isis-wg-mib-06, *Management Information Base for Intermediate System to Intermediate System (IS-IS)*

draft-ietf-mpls-ldp-mib-07, *Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)*

draft-ietf-mpls-lsr-mib-06, *Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base Using SMIv2*

draft-ietf-mpls-te-mib-04, *Multiprotocol Label Switching (MPLS) Traffic Engineering Management Information Base*

draft-ietf-ospf-mib-update-08, *OSPF Version 2 Management Information Base*

draft-ietf-vrrp-unified-mib-06, *Definitions of Managed Objects for the VRRP over IPv4 and IPv6* – IPv6

ESO-CONSORTIUM-MIB revision 200406230000Z, *esoConsortiumMIB*

IANA-ADDRESS-FAMILY-NUMBERS-MIB revision 200203140000Z, *ianaAddressFamilyNumbers*

IANAifType-MIB revision 200505270000Z, *ianaifType*

IANA-RTPROTO-MIB revision 200009260000Z, *ianaRtProtoMIB*

IEEE8021-CFM-MIB revision 200706100000Z, *ieee8021CfmMib*

IEEE8021-PAE-MIB revision 200101160000Z, *ieee8021paeMIB*

IEEE8023-LAG-MIB revision 200006270000Z, *lagMIB*

LLDP-MIB revision 200505060000Z, *lldpMIB*

RFC 1157, *A Simple Network Management Protocol (SNMP)*

RFC 1212, *Concise MIB Definitions*

RFC 1215, *A Convention for Defining Traps for use with the SNMP*

RFC 1724, *RIP Version 2 MIB Extension*

RFC 1901, *Introduction to Community-based SNMPv2*

RFC 2021, *Remote Network Monitoring Management Information Base Version 2 using SMIv2*

RFC 2206, *RSVP Management Information Base using SMIv2*

RFC 2578, *Structure of Management Information Version 2 (SMIv2)*

RFC 2579, *Textual Conventions for SMIv2*

RFC 2580, *Conformance Statements for SMIv2*

RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*

RFC 2819, *Remote Network Monitoring Management Information Base*

RFC 2856, *Textual Conventions for Additional High Capacity Data Types*

RFC 2863, *The Interfaces Group MIB*

RFC 2864, *The Inverted Stack Table Extension to the Interfaces Group MIB*

RFC 2933, *Internet Group Management Protocol MIB*

RFC 3014, *Notification Log MIB*

RFC 3273, *Remote Network Monitoring Management Information Base for High Capacity Networks*

RFC 3410, *Introduction and Applicability Statements for Internet Standard Management Framework*

RFC 3430, *Simple Network Management Protocol (SNMP) over Transmission Control Protocol (TCP) Transport Mapping*

RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*

RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*

RFC 3413, *Simple Network Management Protocol (SNMP) Applications*

RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*

RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*

RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*

RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)* – SNMP over UDP over IPv4

RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

RFC 3419, *Textual Conventions for Transport Addresses*

RFC 3434, *Remote Monitoring MIB Extensions for High Capacity Alarms*

RFC 3584, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*

RFC 3593, *Textual Conventions for MIB Modules Using Performance History Based on 15 Minute Intervals*

RFC 3635, *Definitions of Managed Objects for the Ethernet-like Interface Types*

RFC 3826, *The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model*

RFC 3877, *Alarm Management Information Base (MIB)*

RFC 4001, *Textual Conventions for Internet Network Addresses*

RFC 4022, *Management Information Base for the Transmission Control Protocol (TCP)*

RFC 4113, *Management Information Base for the User Datagram Protocol (UDP)*

RFC 4273, *Definitions of Managed Objects for BGP-4*

RFC 4292, *IP Forwarding Table MIB*

RFC 4293, *Management Information Base for the Internet Protocol (IP)*

RFC 4878, *Definitions and Managed Objects for Operations, Administration, and Maintenance (OAM) Functions on Ethernet-Like Interfaces*

RFC 7420, *Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module*

RFC 7630, *HMAC-SHA-2 Authentication Protocols in the User-based Security Model (USM) for SNMPv3*

7.28 Timing

RFC 3339, *Date and Time on the Internet: Timestamps*

RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*

RFC 8573, *Message Authentication Code for the Network Time Protocol*

7.29 Two-Way Active Measurement Protocol (TWAMP)

RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP)* – server, unauthenticated mode

RFC 5938, *Individual Session Control Feature for the Two-Way Active Measurement Protocol (TWAMP)*

RFC 6038, *Two-Way Active Measurement Protocol (TWAMP) Reflect Octets and Symmetrical Size Features*

RFC 8545, *Well-Known Port Assignments for the One-Way Active Measurement Protocol (OWAMP) and the Two-Way Active Measurement Protocol (TWAMP)* – TWAMP

RFC 8762, *Simple Two-Way Active Measurement Protocol* – unauthenticated

RFC 8972, *Simple Two-Way Active Measurement Protocol Optional Extensions* – unauthenticated

RFC 9503, *Simple Two-Way Active Measurement Protocol (STAMP) Extensions for Segment Routing Networks* – excluding Sections 3, 4.1.2 and 4.1.3

7.30 Virtual Private LAN Service (VPLS)

RFC 4761, *Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling*

RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*

RFC 5501, *Requirements for Multicast Support in Virtual Private LAN Services*

RFC 6074, *Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs)*

RFC 7117, *Multicast in Virtual Private LAN Service (VPLS)*

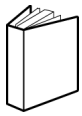
7.31 Yet Another Next Generation (YANG)

RFC 6991, *Common YANG Data Types*

RFC 7950, *The YANG 1.1 Data Modeling Language*

RFC 7951, *JSON Encoding of Data Modeled with YANG*

Customer document and product support



Customer documentation

Customer documentation welcome page

Technical support

Product support portal

Documentation feedback

Customer documentation feedback