Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

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

Junos® OS IPv6 Neighbor Discovery Feature Guide
Copyright © 2019 Juniper Networks, Inc. All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement (“EULA”) posted at https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.
show ipv6 neighbors .................................................. 128
show ipv6 router-advertisement .................................. 131
show log ................................................................. 134
traceroute .............................................................. 138
## List of Figures

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chapter 2</td>
<td>Configuring IPv6 Interfaces and Enabling IPv6 Neighbor Discovery</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td>Figure 1: ICMP Router Discover Topology</td>
<td>22</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Configuring Neighbor Discovery Cache Protection</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>Figure 2: ICMP Router Discover Topology</td>
<td>49</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Troubleshooting</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>Figure 3: Process for Diagnosing Problems in Your Network</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>Figure 4: Network with a Problem</td>
<td>62</td>
</tr>
</tbody>
</table>
List of Tables

About the Documentation ........................................... xi
Table 1: Notice Icons ................................................... xiii
Table 2: Text and Syntax Conventions ............................... xiv

Chapter 6  Troubleshooting ........................................... 61
Table 3: Checklist for Working with Problems on Your Network 61

Chapter 8  Operational Commands ................................. 103
Table 4: Output Control Keys for the monitor interface interface-name Command .............................................. 107
Table 5: Output Control Keys for the monitor interface traffic Command ........................................... 107
Table 6: monitor interface Output Fields .......................... 108
Table 7: monitor start Output Fields ............................... 119
Table 8: show ipv6 neighbors Output Fields ..................... 129
Table 9: show ipv6 router-advertisement Output Fields ........ 132
Table 10: traceroute Output Fields ............................... 140
About the Documentation

- Documentation and Release Notes on page xi
- Using the Examples in This Manual on page xi
- Documentation Conventions on page xiii
- Documentation Feedback on page xv
- Requesting Technical Support on page xv

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at https://www.juniper.net/documentation/.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at https://www.juniper.net/books.

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the load merge or the load merge relative command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a full example. In this case, use the load merge command.

If the example configuration does not start at the top level of the hierarchy, the example is a snippet. In this case, use the load merge relative command. These procedures are described in the following sections.
Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following configuration to a file and name the file `ex-script.conf`. Copy the `ex-script.conf` file to the `/var/tmp` directory on your routing platform.

```plaintext
system {
  scripts {
    commit {
      file ex-script.xsl;
    }
  }
}
```

2. Merge the contents of the file into your routing platform configuration by issuing the `load merge` configuration mode command:

```
[edit]
user@host# load merge /var/tmp/ex-script.conf
load complete
```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

For example, copy the following snippet to a file and name the file `ex-script-snippet.conf`. Copy the `ex-script-snippet.conf` file to the `/var/tmp` directory on your routing platform.

```plaintext
commit {
  file ex-script-snippet.xsl; }
```
2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the `load merge relative` configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the `load` command, see CLI Explorer.

**Documentation Conventions**

Table 1 on page xiii defines notice icons used in this guide.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Informational note" /></td>
<td>Informational note</td>
<td>Indicates important features or instructions.</td>
</tr>
<tr>
<td><img src="image" alt="Caution" /></td>
<td>Caution</td>
<td>Indicates a situation that might result in loss of data or hardware damage.</td>
</tr>
<tr>
<td><img src="image" alt="Warning" /></td>
<td>Warning</td>
<td>Alerts you to the risk of personal injury or death.</td>
</tr>
<tr>
<td><img src="image" alt="Laser warning" /></td>
<td>Laser warning</td>
<td>Alerts you to the risk of personal injury from a laser.</td>
</tr>
<tr>
<td><img src="image" alt="Tip" /></td>
<td>Tip</td>
<td>Indicates helpful information.</td>
</tr>
<tr>
<td><img src="image" alt="Best practice" /></td>
<td>Best practice</td>
<td>Alerts you to a recommended use or implementation.</td>
</tr>
</tbody>
</table>

Table 2 on page xiv defines the text and syntax conventions used in this guide.
<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold text like this</strong></td>
<td>Represents text that you type.</td>
<td>To enter configuration mode, type the <code>configure</code> command:</td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>user@host&gt; configure</code></td>
</tr>
<tr>
<td><strong>Fixed-width text like this</strong></td>
<td>Represents output that appears on the terminal screen.</td>
<td><code>user@host&gt; show chassis alarms</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No alarms currently active</td>
</tr>
<tr>
<td><strong>Italic text like this</strong></td>
<td>• Introduces or emphasizes important new terms.</td>
<td>• A policy term is a named structure that defines match conditions and actions.</td>
</tr>
<tr>
<td></td>
<td>• Identifies guide names.</td>
<td>• Junos OS CLI User Guide</td>
</tr>
<tr>
<td></td>
<td>• Identifies RFC and Internet draft titles.</td>
<td>• RFC 1997, BGP Communities Attribute</td>
</tr>
<tr>
<td><strong>Italic text like this</strong></td>
<td>Represents variables (options for which you substitute a value) in commands or configuration statements.</td>
<td>Configure the machine’s domain name:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[edit]</td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>root@# set system domain-name</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>domain-name</code></td>
</tr>
<tr>
<td><strong>Text like this</strong></td>
<td>Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.</td>
<td>• To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The console port is labeled CONSOLE</td>
</tr>
<tr>
<td><code>&lt; &gt;</code> (angle brackets)</td>
<td>Encloses optional keywords or variables.</td>
<td>stub <code>&lt;default-metric metric&gt;</code>;</td>
</tr>
<tr>
<td></td>
<td>(pipe symbol)</td>
<td>Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(string1</td>
</tr>
<tr>
<td># (pound sign)</td>
<td>Indicates a comment specified on the same line as the configuration statement to which it applies.</td>
<td>rsvp [ # Required for dynamic MPLS only</td>
</tr>
<tr>
<td>[ ] (square brackets)</td>
<td>Encloses a variable for which you can substitute one or more values.</td>
<td>community name members [ community-ids ]</td>
</tr>
<tr>
<td>Indention and braces ( { } )</td>
<td>Identifies a level in the configuration hierarchy.</td>
<td>[edit]</td>
</tr>
<tr>
<td>: (semicolon)</td>
<td>Identifies a leaf statement at a configuration hierarchy level.</td>
<td>routing-options { static { route default { nexthop address; retain; } } }</td>
</tr>
</tbody>
</table>

### GUI Conventions
### Table 2: Text and Syntax Conventions (continued)

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
</table>
| **Bold text like this** | Represents graphical user interface (GUI) items you click or select. | • In the Logical Interfaces box, select *All Interfaces*.  
• To cancel the configuration, click *Cancel*. |
| > (bold right angle bracket) | Separates levels in a hierarchy of menu selections. | In the configuration editor hierarchy, select *Protocols > Ospf*. |

### Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the [Juniper Networks TechLibrary](https://www.juniper.net) site, and do one of the following:

  - Click the thumbs-up icon if the information on the page was helpful to you.
  - Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.

- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

### Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.


- Product warranties—For product warranty information, visit [https://www.juniper.net/support/warranty/](https://www.juniper.net/support/warranty/).

- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.
Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: https://www.juniper.net/customers/support/
- Search for known bugs: https://prsearch.juniper.net/
- Find product documentation: https://www.juniper.net/documentation/
- Find solutions and answer questions using our Knowledge Base: https://kb.juniper.net/
- Download the latest versions of software and review release notes: https://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: https://www.juniper.net/company/communities/
- Create a service request online: https://myjuniper.juniper.net

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://entitlementsearch.juniper.net/entitlementsearch/

Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit https://myjuniper.juniper.net.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see https://support.juniper.net/support/requesting-support/.
Overview

- IPv6 Neighbor Discovery Overview on page 17
- Understanding Secure IPv6 Neighbor Discovery on page 19
- Supported ICMP Router Discovery and IPv6 Neighbor Discovery Standards on page 20

IPv6 Neighbor Discovery Overview

Neighbor discovery is a protocol that allows different nodes on the same link to advertise their existence to their neighbors, and to learn about the existence of their neighbors.

Routers and hosts (nodes) use Neighbor Discovery (ND) messages to determine the link-layer addresses of neighbors that reside on attached links and to overwrite invalid cache entries. Hosts also use ND to find neighboring routers that can forward packets on their behalf.

In addition, nodes use ND to actively track the ability to reach neighbors. When a router (or the path to a router) fails, nodes actively search for alternatives to reach the destination.

This section discusses the following topics:

- Improvements Over Ipv4 Protocols on page 17
- Router Discovery on page 18
- Address Resolution on page 18
- Redirect on page 19

Improvements Over Ipv4 Protocols

IPv6 Neighbor Discovery corresponds to a number of the IPv4 protocols — ARP, ICMP Router Discovery, and ICMP Redirect. However, Neighbor Discovery provides many improvements over the IPv4 set of protocols. These improvements address the following:

- Router discovery—How a host locates routers residing on an attached link.
- Prefix discovery—How a host discovers address prefixes for destinations residing on an attached link. Nodes use prefixes to distinguish between destinations that reside on an attached link and those destinations that it can reach only through a router.
• Parameter discovery—How a node learns various parameters (link parameters or Internet parameters) that it places in outgoing packets.

• Address resolution—How a node uses only a destination IPv6 address to determine a link-layer address for destinations on an attached link.

• Next-hop determination—The algorithm that a node uses for mapping an IPv6 destination address into a neighbor IPv6 address (either the next router hop or the destination itself) to which it plans to send traffic for the destination.

• Neighbor unreachability detection—How a node determines that it can no longer reach a neighbor.

• Duplicate address detection—How a node determines whether an address is already in use by another node.

A router periodically multicasts a router advertisement from each of its multicast interfaces, announcing its availability. Hosts listen for these advertisements for address autoconfiguration and discovery of link-local addresses of the neighboring routers. When a host starts, it multicasts a router solicitation to ask for immediate advertisements.

The router discovery messages do not constitute a routing protocol. They enable hosts to discover the existence of neighboring routers, but are not used to determine which router is best to reach a particular destination.

Neighbor discovery uses the following Internet Control Message Protocol version 6 (ICMPv6) messages: router solicitation, router advertisement, neighbor solicitation, neighbor advertisement, and redirect.

Neighbor discovery for IPv6 replaces the following IPv4 protocols: router discovery (RDISC), Address Resolution Protocol (ARP), and ICMPv4 redirect.

Junos OS Release 9.3 and later supports Secure Neighbor Discovery (SEND). SEND enables you to secure Neighbor Discovery protocol (NDP) messages. It is applicable in environments where physical security on a link is not assured and attacks on NDP messages are a concern. The Junos OS secures NDP messages through cryptographically generated addresses (CGAs).

Router Discovery

Router advertisements can contain a list of prefixes. These prefixes are used for address autoconfiguration, to maintain a database of onlink (on the same data link) prefixes, and for duplication address detection. If a node is onlink, the router forwards packets to that node. If the node is not onlink, the packets are sent to the next router for consideration. For IPv6, each prefix in the prefix list can contain a prefix length, a valid lifetime for the prefix, a preferred lifetime for the prefix, an onlink flag, and an autoconfiguration flag. This information enables address autoconfiguration and the setting of link parameters such as maximum transmission unit (MTU) size and hop limit.

Address Resolution

For IPv6, ICMPv6 neighbor discovery replaces Address Resolution Protocol (ARP) for resolving network addresses to link-level addresses. Neighbor discovery also handles
changes in link-layer addresses, inbound load balancing, anycast addresses, and proxy advertisements.

Nodes requesting the link-layer address of a target node multicast a neighbor solicitation message with the target address. The target sends back a neighbor advertisement message containing its link-layer address.

Neighbor solicitation and advertisement messages are used for detecting duplicate unicast addresses on the same link. Autoconfiguration of an IP address depends on whether there is a duplicate address on that link. Duplicate address detection is a requirement for autoconfiguration.

Neighbor solicitation and advertisement messages are also used for neighbor unreachability detection. Neighbor unreachability detection involves detecting the presence of a target node on a given link.

Redirect

Redirect messages are sent to inform a host of a better next-hop router to a particular destination or an on-link neighbor. This is similar to ICMPv4 redirect. Very similar to the ICMPv4 Redirect feature, the ICMPv6 redirect message is used by routers to inform on-link hosts of a better next-hop for a given destination. The intent is to allow the routers to help hosts make the most efficient local routing decisions possible.

Understanding Secure IPv6 Neighbor Discovery

One of the functions of the IPv6 Neighbor Discovery Protocol (NDP) is to resolve network layer (IP) addresses to link layer (for example, Ethernet) addresses, a function performed in IPv4 by Address Resolution Protocol (ARP). The Secure Neighbor Discovery (SEND) Protocol prevents an attacker who has access to the broadcast segment from abusing NDP or ARP to trick hosts into sending the attacker traffic destined for someone else, a technique known as ARP poisoning.

To protect against ARP poisoning and other attacks against NDP functions, SEND should be deployed where preventing access to the broadcast segment might not be possible.

SEND uses RSA key pairs to produce cryptographically generated addresses, as defined in RFC 3972, Cryptographically Generated Addresses (CGA). This ensures that the claimed source of an NDP message is the owner of the claimed address.
Supported ICMP Router Discovery and IPv6 Neighbor Discovery Standards

Junos OS substantially supports the following RFCs, which define standards for the Internet Control Message Protocol (ICMP for IP version 4 [IPv4]) and neighbor discovery (for IP version 6 [IPv6]).

- RFC 1256, ICMP Router Discovery Messages
- RFC 4861, Neighbor Discovery for IP version 6 (IPv6)
- RFC 2462, IPv6 Stateless Address Autoconfiguration
- RFC 2463, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification
- RFC 4443, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification
- RFC 4861, IPv6 Stateless Address Autoconfiguration
- RFC 4862, Neighbor Discovery for IP version 6 (IPv6)
- RFC 6106, IPv6 Router Advertisement Options for DNS Configuration

Related Documentation
- Supported IPv4, TCP, and UDP Standards
- Supported IPv6 Standards
- Accessing Standards Documents on the Internet
CHAPTER 2

Configuring IPv6 Interfaces and Enabling IPv6 Neighbor Discovery

- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 21
- Example: Configuring Secure IPv6 Neighbor Discovery on page 30

Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery

This example shows how to configure the router or switch to send IPv6 neighbor discovery messages.

- Requirements on page 21
- Overview on page 21
- Configuration on page 23
- Verification on page 25

Requirements

In this example, no special configuration beyond device initialization is required.

Overview

In this example, all of the interfaces in the sample topology are configured with IPv6 addresses. If you plan to extend IPv6 functionality into your LAN, datacenter, or customer networks, you might want to use Stateless Address Auto-Configuration (SLAAC) and that means configuring router advertisements. SLAAC is an IPv6 protocol that provides some similar functionality to DHCP in IPv4. Using SLAAC, network hosts can autoconfigure a globally unique IPv6 address based on the prefix provided by a nearby router in a router advertisement. This removes the need to explicitly configure every interface in a given section of the network. Router advertisement messages are disabled by default, and you must enable them to take advantage of SLAAC.

To configure the router to send router advertisement messages, you must include at least the following statements in the configuration. All other router advertisement configuration statements are optional.

```
protocols {
  router-advertisement {
```
To configure neighbor discovery, include the following statements. You configure router advertisement on a per-interface basis.

```plaintext
protocols {
    router-advertisement {
        interface interface-name {
            current-hop-limit number;
            default-lifetime seconds;
            (link-mtu | no-link-mtu);
            (managed-configuration | no-managed-configuration);
            max-advertisement-interval seconds;
            min-advertisement-interval seconds;
            (other-stateful-configuration | no-other-stateful-configuration);
            prefix prefix {
                (autonomous | no-autonomous);
                (on-link | no-on-link);
                preferred-lifetime seconds;
                valid-lifetime seconds;
            }
            reachable-time milliseconds;
            retransmit-timer milliseconds;
            solicit-router-advertisement-unicast;
            virtual-router-only;
        }
        traceoptions {
            file filename <files number> <size maximum-file-size> <world-readable | no-world-readable>;
            flag flag;
        }
    }
}
```

Figure 1 on page 22 shows a simplified sample topology.

Figure 1: ICMP Router Discover Topology
This example shows how to make sure that all of the IPv6 hosts attached to the subnets in the sample topology can auto-configure a local EUI-64 address.

“CLI Quick Configuration” on page 23 shows the configuration for all of the devices in Figure 1 on page 22. “Step-by-Step Procedure” on page 24 describes the steps on Device R1.

Configuration

**CLI Quick Configuration**

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

**Device R1**

```
set interfaces fe-1/2/0 unit 1 description to-P2
set interfaces fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces fe-1/2/1 unit 5 description to-P4
set interfaces fe-1/2/1 unit 5 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces fe-1/2/2 unit 9 description to-P3
set interfaces fe-1/2/2 unit 9 family inet6 address 2001:db8:0:9::/64 eui-64
set interfaces lo0 unit 1 family inet6 address 2001:db8::1/128
set protocols router-advertisement interface fe-1/2/0.1 prefix 2001:db8:0:1::/64
set protocols router-advertisement interface fe-1/2/1.5 prefix 2001:db8:0:5::/64
set protocols router-advertisement interface fe-1/2/2.9 prefix 2001:db8:0:9::/64
```

**Device R2**

```
set interfaces fe-1/2/0 unit 2 description to-P1
set interfaces fe-1/2/0 unit 2 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces fe-1/2/1 unit 14 description to-P3
set interfaces fe-1/2/1 unit 14 family inet6 address 2001:db8:0:14::/64 eui-64
set interfaces fe-1/2/2 unit 21 description to-P4
set interfaces fe-1/2/2 unit 21 family inet6 address 2001:db8:0:21::/64 eui-64
set interfaces lo0 unit 2 family inet6 address 2001:db8::2/128
set protocols router-advertisement interface fe-1/2/0.2 prefix 2001:db8:0:1::/64
set protocols router-advertisement interface fe-1/2/1.14 prefix 2001:db8:0:14::/64
set protocols router-advertisement interface fe-1/2/2.21 prefix 2001:db8:0:21::/64
```

**Device R3**

```
set interfaces fe-1/2/0 unit 10 description to-P1
set interfaces fe-1/2/0 unit 10 family inet6 address 2001:db8:0:9::/64 eui-64
set interfaces fe-1/2/1 unit 13 description to-P2
set interfaces fe-1/2/1 unit 13 family inet6 address 2001:db8:0:14::/64 eui-64
set interfaces fe-1/2/2 unit 17 description to-P4
set interfaces fe-1/2/2 unit 17 family inet6 address 2001:db8:0:17::/64 eui-64
set interfaces lo0 unit 3 family inet6 address 2001:db8::3/128
set protocols router-advertisement interface fe-1/2/0.10 prefix 2001:db8:0:9::/64
set protocols router-advertisement interface fe-1/2/1.13 prefix 2001:db8:0:14::/64
set protocols router-advertisement interface fe-1/2/2.17 prefix 2001:db8:0:17::/64
```

**Device R4**

```
set interfaces fe-1/2/0 unit 6 description to-P1
set interfaces fe-1/2/0 unit 6 family inet6 address 2001:db8:0:5::/64 eui-64
```
The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure IPv6 neighbor discovery:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 description to-P2
   user@R1# set fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
   user@R1# set fe-1/2/1 unit 5 description to-P4
   user@R1# set fe-1/2/1 unit 5 family inet6 address 2001:db8:0:5::/64 eui-64
   user@R1# set fe-1/2/2 unit 9 description to-P3
   user@R1# set fe-1/2/2 unit 9 family inet6 address 2001:db8:0:9::/64 eui-64
   user@R1# set lo0 unit 1 family inet6 address 2001:db8::1/128
   ```

2. Enable neighbor discovery.

   ```
   [edit protocols router-advertisement]
   user@R1# set interface fe-1/2/0.1 prefix 2001:db8:0:1::/64
   user@R1# set interface fe-1/2/1.5 prefix 2001:db8:0:5::/64
   user@R1# set interface fe-1/2/2.9 prefix 2001:db8:0:9::/64
   ```

3. Results

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

   ```
   user@R1# show interfaces
   fe-1/2/0 {
     unit 1 [
       description to-P2;
       family inet6 [
         address 2001:db8:0:1::/64 {
           eui-64;
         }
       ]
     }
   }
   ```
} fe-1/2/1 {
  unit 5 {
    description to-P4;
    family inet6 {
      address 2001:db8:0:5::/64 {
        eui-64;
      }
    }
  }
} fe-1/2/2 {
  unit 9 {
    description to-P3;
    family inet6 {
      address 2001:db8:0:9::/64 {
        eui-64;
      }
    }
  }
} lo0 {
  unit 1 {
    family inet6 {
      address 2001:db8::1/128;
    }
  }
}

user@R1# show protocols
router-advertisement {
  interface fe-1/2/0.1 {
    prefix 2001:db8:0:1::/64;
  }
  interface fe-1/2/1.5 {
    prefix 2001:db8:0:5::/64;
  }
  interface fe-1/2/2.9 {
    prefix 2001:db8:0:9::/64;
  }
}

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

Verification

To confirm that the configuration is working properly, perform this task:

- Checking the Interfaces on page 26
- Pinging the Interfaces on page 26
- Checking the IPv6 Neighbor Cache on page 27
Checking the Interfaces

**Purpose**
Verify that the interfaces are up, and view the assigned EUI-64 addresses.

**Action**
From operational mode, enter the `show interfaces terse` command.

```plaintext
user@R1> show interfaces terse
```

<table>
<thead>
<tr>
<th>Interface</th>
<th>Admin Link</th>
<th>Proto</th>
<th>Local</th>
<th>Remote</th>
</tr>
</thead>
<tbody>
<tr>
<td>fe-1/2/0</td>
<td>up</td>
<td>up</td>
<td>inet6</td>
<td>2001:db8:0:1:2a0:a514:0:14c/64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fe80::2a0:a514:0:14c/64</td>
</tr>
<tr>
<td>fe-1/2/0.1</td>
<td>up</td>
<td>up</td>
<td>inet6</td>
<td>2001:db8:0:5:2a0:a514:0:54c/64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fe80::2a0:a514:0:54c/64</td>
</tr>
<tr>
<td>fe-1/2/2.9</td>
<td>up</td>
<td>up</td>
<td>inet6</td>
<td>2001:db8:0:9:2a0:a514:0:94c/64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fe80::2a0:a514:0:94c/64</td>
</tr>
<tr>
<td>lo0</td>
<td>up</td>
<td>up</td>
<td>inet6</td>
<td>2001:db8::1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fe80::2a0:a50f:fc56:14c</td>
</tr>
</tbody>
</table>

**Meaning**
The output shows that all interfaces are configured with the IPv6 (inet6) address family. Each IPv6-enabled interface has two IPv6 addresses; one link-local address, and one global address. The global addresses match those shown in Figure 1 on page 22. Junos OS automatically creates a link-local address for any interface that is enabled for IPv6 operation. All link-local addresses begin with the fe80::/64 prefix. The host portion of the address is a full 64 bits long and matches the link-local interface identifier. When an interface address is configured using the `eui-64` statement, its interface identifier matches the interface identifier of the link-local address. This is because link-local addresses are coded according to the EUI-64 specification.

Pinging the Interfaces

**Purpose**
Verify connectivity between the directly connected interfaces.

**Action**
1. Determine the remote router’s IPv6 interface address.

   On Device R2, run the `show interfaces terse` command for the interface that is directly connected to Device R1, and copy the global address into the capture buffer of your terminal emulator.

   ```plaintext
   user@R2> show interfaces fe-1/2/0.2 terse
   ```

<table>
<thead>
<tr>
<th>Interface</th>
<th>Admin Link</th>
<th>Proto</th>
<th>Local</th>
<th>Remote</th>
</tr>
</thead>
<tbody>
<tr>
<td>fe-1/2/0.2</td>
<td>up</td>
<td>up</td>
<td>inet6</td>
<td>2001:db8:0:1:2a0:a514:0:24c/64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fe80::2a0:a514:0:24c/64</td>
</tr>
</tbody>
</table>

2. On Device R1, run the `ping` command, using the global address that you copied.
Meaning

Junos OS uses the same ping command for both IPv4 and IPv6 testing. The lack of any interior gateway protocol (IGP) in the network limits the ping testing to directly-connected neighbors. Repeat the ping test for other directly connected neighbors.

Checking the IPv6 Neighbor Cache

Purpose

Display information about the IPv6 neighbors.

After conducting ping testing, you can find an entries for interface addresses in the IPv6 neighbor cache.

Action

From operational mode, enter the `show ipv6 neighbors` command.

<table>
<thead>
<tr>
<th>IPv6 Address</th>
<th>Linklayer Address</th>
<th>State</th>
<th>Exp Rtr</th>
<th>Secure</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001:db8:0:1:2a0:a514:0:24c</td>
<td>00:05:85:8f:c8:bd</td>
<td>stale</td>
<td>546</td>
<td>yes</td>
</tr>
<tr>
<td>fe-1/2/0.1</td>
<td></td>
<td></td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>fe80::2a0:a514:0:24c</td>
<td>00:05:85:8f:c8:bd</td>
<td>stale</td>
<td>258</td>
<td>yes</td>
</tr>
<tr>
<td>fe-1/2/0.1</td>
<td></td>
<td></td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>fe80::2a0:a514:0:64c</td>
<td>00:05:85:8f:c8:bd</td>
<td>stale</td>
<td>111</td>
<td>yes</td>
</tr>
<tr>
<td>fe-1/2/1.5</td>
<td></td>
<td></td>
<td></td>
<td>no</td>
</tr>
<tr>
<td>fe80::2a0:a514:0:4c</td>
<td>00:05:85:8f:c8:bd</td>
<td>stale</td>
<td>327</td>
<td>yes</td>
</tr>
<tr>
<td>fe-1/2/2.9</td>
<td></td>
<td></td>
<td></td>
<td>no</td>
</tr>
</tbody>
</table>

Meaning

In IPv6, the Address Resolution Protocol (ARP) has been replaced by the Neighbor Discovery Protocol (NDP). The IPv4 command `show arp` is replaced by the IPv6 command `show ipv6 neighbors`. The key pieces of information displayed by this command are the IP address, the MAC (Link Layer) address, and the interface.

Verifying IPv6 Router Advertisements

Purpose

Confirm that devices can be added to the network using SLAAC by ensuring that router advertisements are working properly.

Action

From operational mode, enter the `show ipv6 router-advertisement` command.

```
user@R1> show ipv6 router-advertisement
```

```
Interface: fe-1/2/0.1
Advertisements sent: 37, last sent 00:01:41 ago
```
Meaning  
The output shows that router advertisements are being sent and received on Device R1's interfaces, indicating that both Device R1 and its directly connected neighbors are configured to generate router-advertisements.

Tracing Neighbor Discovery Events

Purpose  
Perform additional validation by tracing router advertisements.
Action 1. Configure trace operations.

```
[edit protocols router-advertisement traceoptions]
user@R1# set file ipv6-nd-trace
user@R1# set traceoptions flag all
user@R1# commit
```

2. Run the `show log` command.

```
user@R1> show log ipv6-nd-trace
Mar 29 14:07:16 trace_on: Tracing to "/var/log/P1/ipv6-nd-trace" started
Mar 29 14:07:16.287229 background dispatch running job
ipv6_ra_delete_interface_config_job for task Router-Advertisement
Mar 29 14:07:16.287452 task_job_delete: delete background job
ipv6_ra_delete_interface_config_job for task Router-Advertisement
Mar 29 14:07:16.287505 background dispatch completed job
ipv6_ra_delete_interface_config_job for task Router-Advertisement
Mar 29 14:07:16.288288 ipv6_ra_iflchange(Router-Advertisement): ifl 0xb904378
  ifl fe-1/2/2.9 104 change 0, intf 0xba140d8
Mar 29 14:07:16.288450 ipv6_ra_iflchange(Router-Advertisement): ifl 0xb904250
  ifl fe-1/2/0.1 85 change 0, intf 0xba14000
Mar 29 14:07:16.288656 ipv6_ra_iflchange(Router-Advertisement): ifl 0xb90444a0
  ifl fe-1/2/1.5 80 change 0, intf 0xba1406c
Mar 29 14:07:16.289293 ipv6_ra_iflchange(Router-Advertisement): ifa 0xba002bc
  fe80::2a0:a514:0:54c ifl fe-1/2/1.5 80 change 0, intf 0xba1406c
Mar 29 14:07:16.289358 -- nochange/add
Mar 29 14:07:16.289624 ipv6_ra_iflchange(Router-Advertisement): ifa 0xba00230
  2001:db8:0:5:2a0:a514:0:54c ifl fe-1/2/1.5 80 change 0, intf 0xba1406c
Mar 29 14:07:16.289682 -- nochange/add
Mar 29 14:07:16.289950 ipv6_ra_iflchange(Router-Advertisement): ifa 0xba001a4
  fe80::2a0:a514:0:14c ifl fe-1/2/0.1 85 change 0, intf 0xba14000
Mar 29 14:07:16.290009 -- nochange/add
Mar 29 14:07:16.290302 ipv6_ra_iflchange(Router-Advertisement): ifa 0xba00118
  2001:db8:0:1:2a0:a514:0:14c ifl fe-1/2/0.1 85 change 0, intf 0xba14000
Mar 29 14:07:16.290365 -- nochange/add
Mar 29 14:07:16.290634 ipv6_ra_iflchange(Router-Advertisement): ifa 0xba003d4
  fe80::2a0:a514:0:94c ifl fe-1/2/2.9 104 change 0, intf 0xba140d8
Mar 29 14:07:16.290694 -- nochange/add
Mar 29 14:07:16.290958 ipv6_ra_iflchange(Router-Advertisement): ifa 0xba00348
  2001:db8:0:9:2a0:a514:0:94c ifl fe-1/2/2.9 104 change 0, intf 0xba140d8
Mar 29 14:07:16.291017 -- nochange/add
Mar 29 14:07:20.808516 task_job_create_foreground: create job ipv6 ra for task
Router-Advertisement
Mar 29 14:07:20.808921 foreground dispatch running job ipv6 ra for task
Router-Advertisement
Mar 29 14:07:20.809027 ipv6_ra_send_advertisement: sending advertisement for
ifl 104 to ff02::1
Mar 29 14:07:20.809087 (4810916) sending advertisement for ifl 104
Mar 29 14:07:20.809170 ifa 0xba00348 2001:db8:0:9:2a0:a514:0:94c/64
Mar 29 14:07:20.809539 --> sent 56 bytes
Mar 29 14:07:20.809660 task_timer_reset: reset Router-Advertisement_ipv6ra
Mar 29 14:07:20.809725 task_timer_set_oneshot_latest: timer
Router-Advertisement_ipv6ra interval set to 7:07
Mar 29 14:07:20.809772 foreground dispatch completed job ipv6 ra for task
Router-Advertisement
```
Example: Configuring Secure IPv6 Neighbor Discovery

This example shows how to configure IPv6 Secure Neighbor Discovery (SEND).

• Requirements on page 30
• Overview on page 30
• Configuration on page 31
• Verification on page 32

Requirements

This example has the following requirements:

• Junos OS Release 9.3 or later
• IPv6 deployed in your network
• If you have not already done so, you must generate or install an RSA key pair.

To generate a new RSA key pair, enter the following command:

```
user@host> request security pki generate-key-pair type rsa certificate-id certificate-id-name size size
```

Overview

To configure SEND, include the following statements:

```
protocols {
    neighbor-discovery {
        onlink-subnet-only;
        secure {
            security-level {
                (default | secure-messages-only);
            }
            cryptographic-address {
                key-length number;
                key-pair pathname;
            }
            timestamp {
                clock-drift number;
                known-peer-window seconds;
                new-peer-window seconds;
            }
            traceoptions {
                file filename <files number> <match regular-expression> <size size> <world-readable | no-world-readable>;
                flag flag;
                no-remote-trace;
            }
        }
    }
}
```
Specify `default` to send and receive both secure and unsecured Neighbor Discovery Protocol (NDP) packets. To configure SEND to accept secured NDP messages only and to drop unsecured ones, specify `secure-messages-only`.

All nodes on the segment need to be configured with SEND if the `secure-messages-only` option is used, which is recommended unless only a small subset of devices require increased protection. Failure to configure SEND for all nodes might result in loss of connectivity.

### Configuration

**CLI Quick Configuration**

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

```plaintext
set protocols neighbor-discovery secure security-level secure-messages-only
set protocols neighbor-discovery secure cryptographic-address key-length 1024
set protocols neighbor-discovery secure cryptographic-address key-pair /var/etc/rsa_key
set protocols neighbor-discovery secure timestamp
```

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a secure IPv6 neighbor discovery:

1. Configure the security level.

   ```plaintext
   [edit protocols neighbor-discovery secure]
   user@host# set security-level secure-messages-only
   ```

2. (Optional) Enable the key length.
   
   The default key length is 1024.
   
   ```plaintext
   [edit protocols neighbor-discovery secure]
   user@host# set cryptographic-address key-length 1024
   ```

3. (Optional) Specify the directory path of the public-private key file generated for the cryptographic address.
   
   The default location of the file is the `/var/etc/rsa_key` directory.
   
   ```plaintext
   [edit protocols neighbor-discovery secure]
   ```
4. (Optional) Configure a timestamp to ensure that solicitation and redirect messages are not being replayed.

   [edit protocols neighbor-discovery secure]
   user@host# set timestamp

   Results  From configuration mode, confirm your configuration by entering the `show protocols` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

      user@host# show protocols neighbor-discovery {
        secure {
          security-level {
            secure-messages-only;
          }
          cryptographic-address {
            key-length 1024;
            key-pair /var/etc/rsa_key;
          }
          timestamp;
        }
      }

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

Verification

Confirm that the configuration is working properly.

- Checking the IPv6 Neighbor Cache on page 32
- Tracing Neighbor Discovery Events on page 33

Checking the IPv6 Neighbor Cache

Purpose  Display information about the IPv6 neighbors.

Action  From operational mode, enter the `show ipv6 neighbors` command.

Meaning  In IPv6, the Address Resolution Protocol (ARP) has been replaced by the NDP. The IPv4 command `show arp` is replaced by the IPv6 command `show ipv6 neighbors`. The key pieces of information displayed by this command are the IP address, the MAC (Link Layer) address, and the interface.
Tracing Neighbor Discovery Events

Purpose  
Perform additional validation by tracing SEND.

Action  
1. Configure trace operations.
   
   [edit protocols neighbor-discovery secure]
   user@host# set traceoptions file send-log
   user@host# set traceoptions flag all

   2. Run the show log command.
   
   user@host> show log send-log
   Apr 11 06:21:26 proto: outgoing pkt on idx 68 does not have CGA (fe80::2a0:a514:0:14c), dropping pkt
   Apr 11 06:26:44 proto: sendd_msg_handler: recv outgoing 96 bytes on idx 70 with offset 40
   Apr 11 06:26:44 dbg: sendd_proto_handler: Modifier (16)
   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
   Apr 11 06:26:44 cga: snd_is_lcl_cga: BEFORE overriding cc, cc:0, ws->col:0
   Apr 11 06:26:44 proto: outgoing pkt on idx 70 does not have CGA (fe80::2a0:a514:0:24c), dropping pkt
   Apr 11 06:26:47 proto: sendd_msg_handler: recv outgoing 96 bytes on idx 68 with offset 40
   Apr 11 06:26:47 dbg: sendd_proto_handler: Modifier (16)
   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00

   Meaning  
The output shows that because the packet does not have a cryptographically generated address, the packet is dropped.

   Related Documentation  
   • Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 21
   • Understanding Secure IPv6 Neighbor Discovery on page 19
   • Understanding IPv6 Neighbor Discovery on page 46
CHAPTER 3

Configuring Neighbor Discovery Protocol Proxy

• Configuring NDP Proxy on page 35

Configuring NDP Proxy

The Neighbor Discovery Protocol (NDP) proxy functionality enables packet forwarding among the hosts that are in the same subnet and are restricted from communicating directly with each other. This functionality is primarily used in a scenario where the proxy node needs to apply access control and intercept traffic flowing among the hosts. When you configure NDP proxy in an SRX Series device, the device sends Neighbor Discovery (ND) advertisements and responds to ND solicitation requests from devices seeking MAC addresses of IPv6 prefixes assigned to hosts inside the SRX Series device.

To configure neighbor discovery proxy on an interface:

1. Set ndp proxy restricted to an interface.

   ```
   set interfaces interface-name family inet6 ndp-proxy interface-restricted
   ```

To disable NDP proxy for an address that is not present in neighbor cache, execute the following command:

```
set protocols neighbor-discovery ndp-proxy no-proxy-on-resolve
```

Use the `show system statistics icmp6` command to get the statistics of events such as NDP proxy requests, NDP proxy conflicts, NDP proxy duplicates, NDP proxy resolve requests and dropped NDP packets.

Related Documentation
CHAPTER 4

Configuring Duplicate Address Detection Proxy

- Configuring DAD Proxy on page 37

Configuring DAD Proxy

The Duplicate Address Detection (DAD) feature detects the usage of duplicate addresses on a local link by using Neighbor Solicitation (NS) messages. The DAD feature is intended for IPv6 address and functions similar to gratuitous ARP in IPv4. The DAD proxy functionality enables an SRX Series device to respond to DAD queries for a node that is prevented from communicating directly with other nodes in the same subnet.

To configure DAD proxy on an interface:

1. Set dad proxy restricted to an interface.
   
   ```
   set interfaces interface-name family inet6 dad-proxy interface-restricted
   ```

To disable DAD proxy for an address that is not present in a neighbor cache, execute the following command:

```
set protocols neighbor-discovery dad-proxy no-proxy-on-resolve
```

Use the `show system statistics icmp6` command to get the statistics of events such as DAD proxy requests, DAD proxy conflicts, DAD proxy duplicates, DAD proxy resolve requests and dropped DAD packets.

Related Documentation
CHAPTER 5

Configuring Neighbor Discovery Cache Protection

- Neighbor Discovery Cache Protection Overview on page 39
- Configuring Neighbor Discovery Cache Protection on page 40
- Example: Configuring Neighbor Discovery Cache Protection to Prevent Denial-of-Service Attacks on page 41
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56

Neighbor Discovery Cache Protection Overview

Routing Engines can be susceptible to certain denial-of-service (DoS) attacks in IPv6 deployment scenarios. IPv6 subnets in general tend to be very large—for example, a /64 subnet might have a high number of unassigned addresses. The control plane of the Routing Engine performs the address resolution for unknown addresses. An attacker can quickly overwhelm the control plane of the Routing Engine by generating resolution requests for this unassigned address space, resulting in a cache overflow. The attacker relies on both the number of requests generated and the rate at which requests are queued up. Such scenarios can tie up router resources and prevent the Routing Engine from answering valid neighbor solicitations and maintaining existing neighbor cache entries, effectively resulting in a DoS attack for legitimate users.

The strategies for mitigating such DoS attacks are as follows:

- Filter unused address space.
- Minimize the size of subnets.
- Configure discard routes for subnets.
- Enforce limits to the size and rate of resolution for entries in the neighbor discovery cache.

Neighbor discovery cache impact can be minimized by restricting the number of IPv6 neighbors and new unresolved next-hop addresses that can be added to the cache. You can set limits per interface by using the `nd6-max-cache` and the `nd6-new-hold-limit` configuration statements or system-wide by using the `nd-system-cache-limit` configuration statement.
NOTE:

- For small sized platforms such as ACX, EX22XX, EX3200, EX33XX, and SRX, default is 20,000.
- For medium sized platforms such as EX4200, EX45XX, EX4300, EX62XX, QFX, and MX, default is 75,000.
- For rest of the platforms, default is 100,000.

Related Documentation
- Configuring Neighbor Discovery Cache Protection on page 40
- Example: Configuring Neighbor Discovery Cache Protection to Prevent Denial-of-Service Attacks on page 41
- nd-system-cache-limit on page 82
- nd6-max-cache on page 83
- nd6-new-hold-limit on page 84

Configuring Neighbor Discovery Cache Protection

Routing Engines can be susceptible to certain types of denial-of-service (DoS) attacks in IPv6 deployment scenarios. IPv6 subnets in general tend be very large; for example, a /64 subnet might have a high number of unassigned addresses. The control plane of the Routing Engine performs the address resolution for unknown addresses. An attacker can quickly overwhelm the control plane of the Routing Engine by generating resolution requests for this unassigned address space, resulting in a cache overflow. An attacker relies on both the number of requests generated and the rate at which requests are queued up.

The neighbor discovery process is that part of the control plane that implements the Neighbor Discovery Protocol. It is responsible for performing address resolution and maintaining the entries in the neighbor cache. One way to mitigate the DoS attacks is by enforcing limits to the size of the neighbor discovery cache and the rate of resolution of new next-hop entries, and by prioritizing certain categories of neighbor discovery traffic. You can configure limits to the neighbor discovery cache per interface and systemwide.

Before you begin, ensure that you are running Junos OS Release 15.1 or later.

Local limits apply to individual interfaces and are defined for resolved and unresolved entries in the neighbor discovery queue, while global limits apply systemwide.

To configure neighbor discovery cache protection on an interface:

1. Configure IPv6 family for the interface.

```
[edit interfaces interface-name unit unit number family]
user@host# set inet6
```
2. Configure the maximum size of the neighbor discovery cache for the interface.

```
[edit interfaces interface-name unit unit number family inet6]
user@host# set nd6-max-cache limit
```

3. Configure the maximum number of unresolved entries in the neighbor discovery cache that can be attached to the interface.

```
[edit interfaces interface-name unit unit number family inet6]
user@host# set nd6-new-hold-limit limit
```

To verify the configuration, execute the `show interfaces interface-name` operational command.

To configure neighbor discovery cache protection systemwide:

- Configure the systemwide limit for the neighbor discovery cache.

```
[edit]
user@host# set system nd-system-cache-limit limit
```

To verify the configured system-wide limits, execute the `show system statistics icmp6` operational command.

**NOTE:**

- For small sized platforms such as ACX, EX22XX, EX3200, EX33XX, and SRX, default is 20,000.
- For medium sized platforms such as EX4200, EX45XX, EX4300, EX62XX, QFX, and MX, default is 75,000.
- For rest of the platforms, default is 100,000.

**Related Documentation**

- Example: Configuring Neighbor Discovery Cache Protection to Prevent Denial-of-Service Attacks on page 41
- IPv6 Neighbor Discovery Overview on page 17
- Neighbor Discovery Cache Protection Overview on page 39

**Example: Configuring Neighbor Discovery Cache Protection to Prevent Denial-of-Service Attacks**

This example shows how to configure a limit to the number of IPv6 neighbor entries that can be added to the neighbor discovery. Enforcing limits to the number of entries in the cache mitigates denial-of-service (DoS) attacks. The neighbor discovery cache feature supports two types of limits:
Local—Local limits are configured per interface and are defined for resolved and unresolved entries in the neighbor discovery cache.

Global—Global limits apply systemwide. A global limit is further defined separately for the public interfaces and management interfaces, for example, fpx0. The management interface has a single global limit and no local limit. The global limit enforces a systemwide cap on entries for the neighbor discovery cache, including for the loopback interface for the internal routing instance, as well as management interfaces and the public interfaces.

- Requirements on page 42
- Overview on page 42
- Configuration on page 42
- Verification on page 44

Requirements

This example requires MX Series routers running Junos OS Release 15.1 or later.

Overview

Routing Engines can be susceptible to certain types of DoS attacks in IPv6 deployment scenarios. IPv6 subnets in general tend to be very large—for example, a /64 subnet might have a high number of unassigned addresses, which can be used to perform DoS attacks. The control plane of the Routing Engine performs the address resolution for unknown addresses. An attacker can quickly overwhelm the control plane of the Routing Engine by generating resolution requests for this unassigned address space and overflow the queue. The attacker relies on both the number of requests generated and the rate at which requests are queued up.

The neighbor discovery process is that part of the control plane that implements the Neighbor Discovery Protocol. It is responsible for performing address resolution and maintaining the neighbor cache. One way to mitigate DoS attacks is by enforcing limits on the neighbor discovery queue limits, which can be done by restricting queue size and the rate of resolution, and by prioritizing certain categories of neighbor discovery traffic.

Configuration

To configure neighbor discovery cache protection, perform these tasks:

- Configuring Neighbor Discovery Cache Protection on page 43
- Results on page 43

CLI Quick Configuration

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

```
set interfaces ge-0/3/0 unit 5 family inet6 nd6-max-cache 100
set interfaces ge-0/3/0 unit 5 family inet6 nd6-new-hold-limit 100
```
You can also configure a systemwide limit to the number of IPv6 neighbor entries in the neighbor discovery cache. This limit also includes the loopback interface, management interfaces, and the public interfaces.

```
set system nd-system-cache-limit 100
```

The limit distribution from the `nd-system-cache-limit` statement for different interface types is performed according to certain fixed percentages. When `nd-system-cache-limit` is defined as $X$ and the internal routing interface neighbor discovery cache limit is $Y$ (default is 200), then:

- Public maximum cache limit, $Z = 80\% \text{ of } (X - Y)$
- Management interface maximum cache limit (for example, fxp0), $M = 20\% \text{ of } (X-Y)$

### Configuring Neighbor Discovery Cache Protection

**Step-by-Step Procedure**

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure neighbor discovery cache protection per interface:

- Configure the `nd6-max-cache` and `nd6-new-hold-limit`.

```
[edit]
user@host# set interfaces ge-0/3/0 unit 5 family inet6 nd6-max-cache 100
user@host# set interfaces ge-0/3/0 unit 5 family inet6 nd6-new-hold-limit 100
```

### Results

To confirm neighbor discovery cache protection locally, enter `show interfaces ge-0/3/0` from configuration mode. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@host# show interfaces ge-0/3/0
unit 5{
    family inet6 {
        nd6-max-cache 100;
        nd6-new-hold-limit 100;
    }
}
```
Verification

Confirm that the configuration is working properly.

- Verifying Neighbor Discovery Cache Protection Globally on page 44
- Verifying Neighbor Discovery Cache Protection Locally on page 45

Verifying Neighbor Discovery Cache Protection Globally

Purpose

Verify that the output reflects the systemwide limit for the neighbor discovery cache.

Action

From operational mode, run the `show system statistics icmp6` command.

```
user@host> show system statistics icmp6
icmp6:
  79 Calls to icmp_error
  0 Errors not generated because old message was icmp error
  0 Errors not generated because rate limitation
Output histogram:
  79 unreach
  30 echo
  163 multicast listener query
  6 multicast listener report
  940 neighbor solicitation
  694184 neighbor advertisement
  0 Messages with bad code fields
  0 Messages < minimum length
  0 Bad checksums
  0 Messages with bad length
Input histogram:
  10 echo reply
  6 multicast listener report
  693975 neighbor solicitation
Histogram of error messages to be generated:
  0 No route
  0 Administratively prohibited
  0 Beyond scope
  79 Address unreachable
  0 Port unreachable
  0 Time exceed transit
  0 Time exceed reassembly
  0 Erroneous header field
  0 Unrecognized next header
  0 Unrecognized option
  0 Unknown
  0 Message responses generated
  0 Messages with too many ND options
  100000 Max System ND nh cache limit
  79840 Max Public ND nh cache limit
  200 Max IRI ND nh cache limit
  19960 Max Management intf ND nh cache limit
  79840 Current Public ND nexthops present
  4 Current IRI ND nexthops present
  0 Current Management ND nexthops present
  909266 Total ND nexthops creation failed as limit reached
```
Meaning  The systemwide cap enforced on the neighbor discovery cache entries is 100000.

Management ND nexthops creation failed as mgt limit reached indicates the drop count for the management interface when the systemwide limit is reached. Total ND nexthops creation failed as limit reached indicates failure for management, public, or Internal routing instance interfaces, and Public ND nexthops creation failed as public limit reached indicates the drop count for public interfaces when the systemwide limit to the number of entries is reached.

Verifying Neighbor Discovery Cache Protection Locally

Purpose  Verify that the output reflects the configured interface limits.

Action  From operational mode, run the show interfaces ge-0/3/0 command.

user@host> show interfaces ge-0/3/0

Logical interface ge-0/2/0.8 (Index 348) (SNMP ifIndex 690)
  Flags: Up SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.8 ] Encapsulation: ENET2
  Input packets : 181628
  Output packets: 79872
  Protocol inet6, MTU: 1500
  Max nh cache: 100000, New hold nh limit: 100000, Curr nh cnt: 79840, Curr new hold cnt: 0, NH drop cnt: 0
  Flags: Is-Primary
  Addresses, Flags: Is-Preferred Is-Primary
    Destination: 8001:1::/64, Local: 8001:1::1:1
    Addresses, Flags: Is-Preferred
    Destination: fe80::/64, Local: fe80::56e0:3200:8c6:e0a4
  Protocol multiservice, MTU: Unlimited

Meaning  The maximum number of total entries and the maximum number of entries for new unresolved next-hop addresses that can be attached to interface ge-0/3/0 is 100000.

NH drop cnt refers to the number of neighbor discovery requests not serviced because the interface maximum queue size limits have been reached.

Related Documentation  • Configuring Neighbor Discovery Cache Protection on page 40
  • IPv6 Neighbor Discovery Overview on page 17
  • nd-system-cache-limit on page 82
  • nd6-max-cache on page 83
  • nd6-new-hold-limit on page 84
Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery

- Understanding IPv6 Neighbor Discovery on page 46
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 47

Understanding IPv6 Neighbor Discovery

IPv6 Neighbor Discovery has many improvements when compared to the corresponding IPv4 protocols.

For instance, Neighbor Discovery moves address resolution to the ICMP layer, which makes it much less media dependent than ARP, as well as adding the ability to use IP layer security when needed.

Additionally, Neighbor Discovery uses link-local addresses. This allows all nodes to maintain their router associations even when the site is renumbered to a new global prefix.

Another improvement worth noting is that Neighbor Discovery messages carry link-layer address information, so a single message (or pair of messages) is all that is needed for nodes to resolve the others’ addresses. No additional address resolution is needed.

Neighbor unreachability detection is built in, making packet delivery much more robust in a changing network. Using neighbor unreachability detection, Neighbor Discovery detects router failures, link failures, and partial link failures such as one-way communication.

And finally, IPv6 router advertisements carry prefixes (including network masks) and support multiple prefixes on the same link. Hosts can learn on-link prefixes from router advertisements or, when the router is configured to withhold them, from redirects as needed.

SLAAC

In addition to all the other improvements it brings to the networking world, Neighbor Discovery also enables address autoconfiguration, namely Stateless Address Autoconfiguration (SLAAC). IPv6 maintains the capability for stateful address assignment through DHCPv6 (and static assignment), but SLAAC provides a lightweight address configuration method that might be desirable in many circumstances.

SLAAC provides plug-and-play IP connectivity in two phases: Phase 1: Link-local address assignment; and then, in Phase 2: Global address assignment.

- Phase 1—Steps for local connectivity:
  1. Link-Local Address Generation: Any time that a multicast-capable IPv6-enabled interface is turned up, the node generates a link-local address for that interface. This is done by appending an interface identifier to the link-local prefix (FE80::/10). The auto generated link-local address cannot be deleted. However, a new link-local address can also be manually entered, which overwrites the auto generated link-local address.
2. **Duplicate Detection:** Before assigning the new link-local address to its interface, the node verifies that the address is unique. This is accomplished by sending a Neighbor Solicitation message destined to the new address. If there is a reply, then the address is a duplicate and the process stops, requiring operator intervention.

3. **Link-Local Address Assignment:** If the address is unique, the node assigns it to the interface for which it was generated.

At this point, the node has IPv6 connectivity to all other nodes on the same link. Phase 2 can only be completed by hosts. The router’s interface addresses must be configured by other means.

**Phase 2—Steps for global connectivity:**

1. **Router Advertisement:** The node sends a Router Solicitation to prompt all on-link routers to send it router advertisements. When the router is enabled to provide stateless autoconfiguration support, the router advertisement contains a subnet prefix for use by neighboring hosts.

2. **Global Address Generation:** Once it receives a subnet prefix from a router, the host generates a global address by appending the interface id to the supplied prefix.

3. **Duplicate Address Detection:** The host again performs Duplicate Address Detection (DAD), this time for the new global address. **4. Global Address Assignment:** Assuming that the address is not a duplicate, the host assigns it to the interface.

This process ensures full IPv6 global connectivity with no manual host configuration and very little router configuration.

**See Also**  
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 21

**Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery**

This example shows how to configure the router or switch to send IPv6 neighbor discovery messages.

- Requirements on page 47
- Overview on page 47
- Configuration on page 49
- Verification on page 52

**Requirements**

In this example, no special configuration beyond device initialization is required.

**Overview**

In this example, all of the interfaces in the sample topology are configured with IPv6 addresses. If you plan to extend IPv6 functionality into your LAN, datacenter, or customer networks, you might want to use Stateless Address Auto-Configuration (SLAAC) and that means configuring router advertisements. SLAAC is an IPv6 protocol that provides some similar functionality to DHCP in IPv4. Using SLAAC, network hosts can autoconfigure
a globally unique IPv6 address based on the prefix provided by a nearby router in a router advertisement. This removes the need to explicitly configure every interface in a given section of the network. Router advertisement messages are disabled by default, and you must enable them to take advantage of SLAAC.

To configure the router to send router advertisement messages, you must include at least the following statements in the configuration. All other router advertisement configuration statements are optional.

```
protocols {
  router-advertisement {
    interface interface-name {
      prefix prefix;
    }
  }
}
```

To configure neighbor discovery, include the following statements. You configure router advertisement on a per-interface basis.

```
protocols {
  router-advertisement {
    interface interface-name {
      current-hop-limit number;
      default-lifetime seconds;
      (link-mtu | no-link-mtu);
      (managed-configuration | no-managed-configuration);
      max-advertisement-interval seconds;
      min-advertisement-interval seconds;
      (other-stateful-configuration | no-other-stateful-configuration);
      prefix prefix {
        (autonomous | no-autonomous);
        (on-link | no-on-link);
        preferred-lifetime seconds;
        valid-lifetime seconds;
      }
      reachable-time milliseconds;
      retransmit-timer milliseconds;
      solicit-router-advertisement-unicast;
      virtual-router-only;
    }
  }
  traceoptions {
    file filename <files number> <size maximum-file-size> <world-readable | no-world-readable>;
    flag flag;
  }
}
```

Figure 1 on page 22 shows a simplified sample topology.
This example shows how to make sure that all of the IPv6 hosts attached to the subnets in the sample topology can auto-configure a local EUI-64 address.

“CLI Quick Configuration” on page 23 shows the configuration for all of the devices in Figure 1 on page 22. “Step-by-Step Procedure” on page 24 describes the steps on Device R1.

**Configuration**

### CLI Quick Configuration

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

**Device R1**

```text
set interfaces fe-1/2/0 unit 1 description to-P2
set interfaces fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces fe-1/2/1 unit 5 description to-P4
set interfaces fe-1/2/1 unit 5 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces fe-1/2/2 unit 9 description to-P3
set interfaces fe-1/2/2 unit 9 family inet6 address 2001:db8:0:9::/64 eui-64
set interfaces lo0 unit 1 family inet6 address 2001:db8::1/128
set protocols router-advertisement interface fe-1/2/0.1 prefix 2001:db8:0:1::/64
set protocols router-advertisement interface fe-1/2/1.14 prefix 2001:db8:0:14::/64
set protocols router-advertisement interface fe-1/2/2.21 prefix 2001:db8:0:21::/64
```

**Device R2**

```text
set interfaces fe-1/2/0 unit 2 description to-P1
set interfaces fe-1/2/0 unit 2 family inet6 address 2001:db8:0:1::/64 eui-64
set interfaces fe-1/2/1 unit 14 description to-P3
set interfaces fe-1/2/1 unit 14 family inet6 address 2001:db8:0:14::/64 eui-64
set interfaces fe-1/2/2 unit 21 description to-P4
set interfaces fe-1/2/2 unit 21 family inet6 address 2001:db8:0:21::/64 eui-64
set interfaces lo0 unit 2 family inet6 address 2001:db8::2/128
set protocols router-advertisement interface fe-1/2/0.2 prefix 2001:db8:0:1::/64
set protocols router-advertisement interface fe-1/2/1.14 prefix 2001:db8:0:14::/64
set protocols router-advertisement interface fe-1/2/2.21 prefix 2001:db8:0:21::/64
```

**Device R3**

```text
set interfaces fe-1/2/0 unit 10 description to-P1
```
set interfaces fe-1/2/0 unit 10 family inet6 address 2001:db8:0:9::/64 eui-64
set interfaces fe-1/2/1 unit 13 description to-P2
set interfaces fe-1/2/1 unit 13 family inet6 address 2001:db8:0:14::/64 eui-64
set interfaces fe-1/2/2 unit 17 description to-P4
set interfaces fe-1/2/2 unit 17 family inet6 address 2001:db8:0:17::/64 eui-64
set interfaces lo0 unit 3 family inet6 address 2001:db8::3/128
set protocols router-advertisement interface fe-1/2/0.10 prefix 2001:db8:0:9::/64
set protocols router-advertisement interface fe-1/2/1.13 prefix 2001:db8:0:14::/64
set protocols router-advertisement interface fe-1/2/2.17 prefix 2001:db8:0:17::/64

Device R4

set interfaces fe-1/2/0 unit 6 description to-P1
set interfaces fe-1/2/0 unit 6 family inet6 address 2001:db8:0:5::/64 eui-64
set interfaces fe-1/2/1 unit 18 description to-P3
set interfaces fe-1/2/1 unit 18 family inet6 address 2001:db8:0:17::/64 eui-64
set interfaces fe-1/2/2 unit 22 description to-P2
set interfaces fe-1/2/2 unit 22 family inet6 address 2001:db8:0:21::/64 eui-64
set interfaces lo0 unit 4 family inet6 address 2001:db8::4/128
set protocols router-advertisement interface fe-1/2/0.6 prefix 2001:db8:0:5::/64
set protocols router-advertisement interface fe-1/2/1.18 prefix 2001:db8:0:17::/64
set protocols router-advertisement interface fe-1/2/2.22 prefix 2001:db8:0:21::/64

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see Using the CLI Editor in Configuration Mode in the CLI User Guide.

To configure a IPv6 neighbor discovery:

1. Configure the network interfaces.

   This example shows multiple loopback interface addresses to simulate attached networks.

   ```
   [edit interfaces]
   user@R1# set fe-1/2/0 unit 1 description to-P2
   user@R1# set fe-1/2/0 unit 1 family inet6 address 2001:db8:0:1::/64 eui-64
   user@R1# set fe-1/2/1 unit 5 description to-P4
   user@R1# set fe-1/2/1 unit 5 family inet6 address 2001:db8:0:5::/64 eui-64
   user@R1# set fe-1/2/2 unit 9 description to-P3
   user@R1# set fe-1/2/2 unit 9 family inet6 address 2001:db8:0:9::/64 eui-64
   user@R1# set lo0 unit 1 family inet6 address 2001:db8::1/128
   ```

2. Enable neighbor discovery.

   ```
   [edit protocols router-advertisement]
   user@R1# set interface fe-1/2/0.1 prefix 2001:db8:0:1::/64
   user@R1# set interface fe-1/2/1.5 prefix 2001:db8:0:5::/64
   user@R1# set interface fe-1/2/2.9 prefix 2001:db8:0:9::/64
   ```
Results From configuration mode, confirm your configuration by entering the `show interfaces` and `show protocols` commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```bash
user@R1# show interfaces
fe-1/2/0 {
  unit 1 [
    description to-P2;
    family inet6 {
      address 2001:db8:0:1::/64 {
        eui-64;
      }
    }
  }
}
fe-1/2/1 {
  unit 5 [
    description to-P4;
    family inet6 {
      address 2001:db8:0:5::/64 {
        eui-64;
      }
    }
  }
}
fe-1/2/2 {
  unit 9 [
    description to-P3;
    family inet6 {
      address 2001:db8:0:9::/64 {
        eui-64;
      }
    }
  }
}
lo0 {
  unit 1 [
    family inet6 {
      address 2001:db8::1/128;
    }
  }
}

user@R1# show protocols
router-advertisement {
  interface fe-1/2/0.1 {
    prefix 2001:db8:0:1::/64;
  }
  interface fe-1/2/1.5 {
    prefix 2001:db8:0:5::/64;
  }
  interface fe-1/2/2.9 {
    prefix 2001:db8:0:9::/64;
  }
}
```
If you are done configuring the device, enter `commit` from configuration mode.

**Verification**

To confirm that the configuration is working properly, perform this task:

- Checking the Interfaces on page 52
- Pinging the Interfaces on page 52
- Checking the IPv6 Neighbor Cache on page 53
- Verifying IPv6 Router Advertisements on page 54
- Tracing Neighbor Discovery Events on page 55

**Checking the Interfaces**

**Purpose**
Verify that the interfaces are up, and view the assigned EUI-64 addresses.

**Action**
From operational mode, enter the `show interfaces terse` command.

```
user@R1> show interfaces terse

Interface     Admin Link Proto    Local                  Remote
fe-1/2/0      up    up   inet6    2001:db8:0:1:2a0:a514:0:14c/64
               fe80::2a0:a514:0:14c/64
fe-1/2/0.1    up    up   inet6    2001:db8:0:5:2a0:a514:0:54c/64
               fe80::2a0:a514:0:54c/64
fe-1/2/1.5    up    up   inet6    2001:db8:0:9:2a0:a514:0:94c/64
               fe80::2a0:a514:0:94c/64
fe-1/2/2.9    up    up   inet6    2001:db8::1
               fe80::2a0:a50f:fc56:14c
lo0           up    up   inet6    2001:db8::1
               fe80::2a0:a50f:fc56:14c
lo0.1         up    up   inet6
```

**Meaning**
The output shows that all interfaces are configured with the IPv6 (inet6) address family. Each IPv6-enabled interface has two IPv6 addresses; one link-local address, and one global address. The global addresses match those shown in Figure 1 on page 22. Junos OS automatically creates a link-local address for any interface that is enabled for IPv6 operation. All link-local addresses begin with the fe80::/64 prefix. The host portion of the address is a full 64 bits long and matches the link-local interface identifier. When an interface address is configured using the `eui-64` statement, its interface identifier matches the interface identifier of the link-local address. This is because link-local addresses are coded according to the EUI-64 specification.

**Pinging the Interfaces**

**Purpose**
Verify connectivity between the directly connected interfaces.
**Action**

1. Determine the remote router’s IPv6 interface address.

   On Device R2, run the `show interfaces terse` command for the interface that is directly connected to Device R1, and copy the global address into the capture buffer of your terminal emulator.

   ```
   user@R2> show interfaces fe-1/2/0.2 terse
   Interface               Admin Link Proto    Local               Remote
   fe-1/2/0.2              up    up   inet6    2001:db8:0:1:2a0:a514:0:24c/64
   fe80::2a0:a514:0:24c/64
   ```

2. On Device R1, run the `ping` command, using the global address that you copied.

   ```
   user@R1> ping 2001:db8:0:1:2a0:a514:0:24c
   PING6(56=40+8+8 bytes) 2001:db8:0:1:2a0:a514:0:14c -->
   2001:db8:0:1:2a0:a514:0:24c
   16 bytes from 2001:db8:0:1:2a0:a514:0:24c, icmp_seq=0 hlim=64 time=20.412 ms
   16 bytes from 2001:db8:0:1:2a0:a514:0:24c, icmp_seq=1 hlim=64 time=18.897 ms
   16 bytes from 2001:db8:0:1:2a0:a514:0:24c, icmp_seq=2 hlim=64 time=1.389 ms
   ```

**Meaning**

Junos OS uses the same ping command for both IPv4 and IPv6 testing. The lack of any interior gateway protocol (IGP) in the network limits the ping testing to directly-connected neighbors. Repeat the ping test for other directly connected neighbors.

### Checking the IPv6 Neighbor Cache

**Purpose**

Display information about the IPv6 neighbors.

After conducting ping testing, you can find an entries for interface addresses in the IPv6 neighbor cache.

**Action**

From operational mode, enter the `show ipv6 neighbors` command.

```
user@R1> show ipv6 neighbors
IPv6 Address                 Linklayer Address  State       Exp Rtr Secure
Interface 00:05:85:8f:cd:bd
fe-1/2/0.1 stale 546 yes no
fe80::2a0:a514:0:24c/64 stale 258 yes no
fe80::2a0:a514:0:64c/64 stale 111 yes no
fe80::2a0:a514:0:4c/64 stale 327 yes no
```

**Meaning**

In IPv6, the Address Resolution Protocol (ARP) has been replaced by the Neighbor Discovery Protocol (NDP). The IPv4 command `show arp` is replaced by the IPv6 command `show ipv6 neighbors`. The key pieces of information displayed by this command are the IP address, the MAC (Link Layer) address, and the interface.
Verifying IPv6 Router Advertisements

Purpose  Confirm that devices can be added to the network using SLAAC by ensuring that router advertisements are working properly.

Action  From operational mode, enter the `show ipv6 router-advertisement` command.

```
user@R1> show ipv6 router-advertisement

Interface: fe-1/2/0.1
  Advertisements sent: 37, last sent 00:01:41 ago
  Solicits received: 0
  Advertisements received: 38
  Advertisement from fe80::2a0:a514:0:24c, heard 00:05:46 ago
    Managed: 0
    Other configuration: 0
    Reachable time: 0 ms
    Default lifetime: 1800 sec
    Retransmit timer: 0 ms
    Current hop limit: 64
    Prefix: 2001:db8:0:1::/64
      Valid lifetime: 2592000 sec
      Preferred lifetime: 604800 sec
    On link: 1
    Autonomous: 1

Interface: fe-1/2/1.5
  Advertisements sent: 36, last sent 00:05:49 ago
  Solicits received: 0
  Advertisements received: 37
  Advertisement from fe80::2a0:a514:0:64c, heard 00:00:54 ago
    Managed: 0
    Other configuration: 0
    Reachable time: 0 ms
    Default lifetime: 1800 sec
    Retransmit timer: 0 ms
    Current hop limit: 64
    Prefix: 2001:db8:0:5::/64
      Valid lifetime: 2592000 sec
      Preferred lifetime: 604800 sec
    On link: 1
    Autonomous: 1

Interface: fe-1/2/2.9
  Advertisements sent: 36, last sent 00:01:37 ago
  Solicits received: 0
  Advertisements received: 38
  Advertisement from fe80::2a0:a514:0:a4c, heard 00:01:00 ago
    Managed: 0
    Other configuration: 0
    Reachable time: 0 ms
    Default lifetime: 1800 sec
    Retransmit timer: 0 ms
    Current hop limit: 64
    Prefix: 2001:db8:0:9::/64
      Valid lifetime: 2592000 sec
      Preferred lifetime: 604800 sec
    On link: 1
    Autonomous: 1
```
Meaning  The output shows that router advertisements are being sent and received on Device R1's interfaces, indicating that both Device R1 and its directly connected neighbors are configured to generate router-advertisements.

**Tracing Neighbor Discovery Events**

**Purpose**  Perform additional validation by tracing router advertisements.

**Action**  
1. Configure trace operations.

   ```
   [edit protocols router-advertisement traceoptions]
   user@R1# set file ipv6-nd-trace
   user@R1# set traceoptions flag all
   user@R1# commit
   ```

   2. Run the **show log** command.

   ```
   user@R1> show log ipv6-nd-trace
   ```

   ```
   Mar 29 14:07:16 trace_on: Tracing to "/var/log/P1/ipv6-nd-trace" started
   Mar 29 14:07:16.287229 background dispatch running job
   ipv6_ra_delete_interface_config_job for task Router-Advertisement
   Mar 29 14:07:16.287452 task_job_delete: delete background job
   ipv6_ra_delete_interface_config_job for task Router-Advertisement
   Mar 29 14:07:16.287505 background dispatch completed job
   ipv6_ra_delete_interface_config_job for task Router-Advertisement
   Mar 29 14:07:16.288288 ipv6_ra_ifachange(Router-Advertisement): ifl 0xb904378
   ifl fe-1/2/2.9 104 change 0, intf 0xba140d8
   Mar 29 14:07:16.288450 ipv6_ra_ifachange(Router-Advertisement): ifl 0xb904250
   ifl fe-1/2/0.1 85 change 0, intf 0xba14000
   Mar 29 14:07:16.288566 ipv6_ra_ifachange(Router-Advertisement): ifl 0xb9044a0
   ifl fe-1/2/1.5 80 change 0, intf 0xba1406c
   Mar 29 14:07:16.289293 ipv6_ra_ifachange(Router-Advertisement): ifa 0xba002bc
   fe80::2a0:a514:0:54c ifl fe-1/2/1.5 80 change 0, intf 0xba1406c
   Mar 29 14:07:16.289358 -- nochange/add
   Mar 29 14:07:16.289624 ipv6_ra_ifachange(Router-Advertisement): ifa 0xba00230
   2001:db8:0:5:2a0:a514:0:54c ifl fe-1/2/1.5 80 change 0, intf 0xba1406c
   Mar 29 14:07:16.289682 -- nochange/add
   Mar 29 14:07:16.289950 ipv6_ra_ifachange(Router-Advertisement): ifa 0xba001a4
   fe80::2a0:a514:0:14c ifl fe-1/2/0.1 85 change 0, intf 0xba14000
   Mar 29 14:07:16.290009 -- nochange/add
   Mar 29 14:07:16.290302 ipv6_ra_ifachange(Router-Advertisement): ifa 0xba00118
   2001:db8:0:1:2a0:a514:0:14c ifl fe-1/2/0.1 85 change 0, intf 0xba14000
   Mar 29 14:07:16.290365 -- nochange/add
   Mar 29 14:07:16.290634 ipv6_ra_ifachange(Router-Advertisement): ifa 0xba003d4
   fe80::2a0:a514:0:94c ifl fe-1/2/2.9 104 change 0, intf 0xba140d8
   Mar 29 14:07:16.290694 -- nochange/add
   Mar 29 14:07:16.290958 ipv6_ra_ifachange(Router-Advertisement): ifa 0xba00348
   2001:db8:0:9:2a0:a514:0:94c ifl fe-1/2/2.9 104 change 0, intf 0xba140d8
   Mar 29 14:07:16.291017 -- nochange/add
   Mar 29 14:07:20.808516 task_job_create_foreground: create job ipv6 ra for task
   Router-Advertisement
   Mar 29 14:07:20.808921 foreground dispatch running job ipv6 ra for task
   Router-Advertisement
   Mar 29 14:07:20.809027 ipv6_ra_send_advertisement: sending advertisement for
   ifl 104 to ff02::1
   Mar 29 14:07:20.809087 (4810916) sending advertisement for ifl 104
   ```
See Also

- IPv6 Neighbor Discovery Overview on page 17

Example: Configuring Secure IPv6 Neighbor Discovery

- Understanding Secure IPv6 Neighbor Discovery on page 56
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56

Understanding Secure IPv6 Neighbor Discovery

One of the functions of the IPv6 Neighbor Discovery Protocol (NDP) is to resolve network layer (IP) addresses to link layer (for example, Ethernet) addresses, a function performed in IPv4 by Address Resolution Protocol (ARP). The Secure Neighbor Discovery (SEND) Protocol prevents an attacker who has access to the broadcast segment from abusing NDP or ARP to trick hosts into sending the attacker traffic destined for someone else, a technique known as ARP poisoning.

To protect against ARP poisoning and other attacks against NDP functions, SEND should be deployed where preventing access to the broadcast segment might not be possible.

SEND uses RSA key pairs to produce cryptographically generated addresses, as defined in RFC 3972, Cryptographically Generated Addresses (CGA). This ensures that the claimed source of an NDP message is the owner of the claimed address.

See Also

- Example: Configuring Secure IPv6 Neighbor Discovery on page 30

Example: Configuring Secure IPv6 Neighbor Discovery

This example shows how to configure IPv6 Secure Neighbor Discovery (SEND).

- Requirements on page 56
- Overview on page 57
- Configuration on page 57
- Verification on page 59

Requirements

This example has the following requirements:

- Junos OS Release 9.3 or later
- IPv6 deployed in your network
If you have not already done so, you must generate or install an RSA key pair.

To generate a new RSA key pair, enter the following command:

```
user@host> request security pki generate-key-pair type rsa certificate-id certificate-id-name size size
```

**Overview**

To configure SEND, include the following statements:

```plaintext
protocols {
    neighbor-discovery {
        onlink-subnet-only;
        secure {
            security-level {
                (default | secure-messages-only);
            }
            cryptographic-address {
                key-length number;
                key-pair pathname;
            }
            timestamp {
                clock-drift number;
                known-peer-window seconds;
                new-peer-window seconds;
            }
            traceoptions {
                file filename <files number> <match regular-expression> <size size>
                <world-readable | no-world-readable>;
                flag flag;
                no-remote-trace;
            }
        }
    }
}
```

Specify `default` to send and receive both secure and unsecured Neighbor Discovery Protocol (NDP) packets. To configure SEND to accept secured NDP messages only and to drop unsecured ones, specify `secure-messages-only`.

All nodes on the segment need to be configured with SEND if the `secure-messages-only` option is used, which is recommended unless only a small subset of devices require increased protection. Failure to configure SEND for all nodes might result in loss of connectivity.

**Configuration**

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.
To configure a secure IPv6 neighbor discovery:

1. Configure the security level.
   
   ```
   [edit protocols neighbor-discovery secure]
   user@host# set security-level secure-messages-only
   ```

2. (Optional) Enable the key length.
   
   The default key length is 1024.
   
   ```
   [edit protocols neighbor-discovery secure]
   user@host# set cryptographic-address key-length 1024
   ```

3. (Optional) Specify the directory path of the public-private key file generated for the cryptographic address.
   
   The default location of the file is the `/var/etc/rsa_key` directory.
   
   ```
   [edit protocols neighbor-discovery secure]
   user@host# set cryptographic-address key-pair /var/etc/rsa_key
   ```

4. (Optional) Configure a timestamp to ensure that solicitation and redirect messages are not being replayed.
   
   ```
   [edit protocols neighbor-discovery secure]
   user@host# set timestamp
   ```

Results

From configuration mode, confirm your configuration by entering the `show protocols` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
key-length 1024;
key-pair /var/etc/rsa_key;
} timestamp;
}

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

**Verification**

Confirm that the configuration is working properly.

- Checking the IPv6 Neighbor Cache on page 59
- Tracing Neighbor Discovery Events on page 59

**Checking the IPv6 Neighbor Cache**

**Purpose**

Display information about the IPv6 neighbors.

**Action**

From operational mode, enter the `show ipv6 neighbors` command.

**Meaning**

In IPv6, the Address Resolution Protocol (ARP) has been replaced by the NDP. The IPv4 command `show arp` is replaced by the IPv6 command `show ipv6 neighbors`. The key pieces of information displayed by this command are the IP address, the MAC (Link Layer) address, and the interface.

**Tracing Neighbor Discovery Events**

**Purpose**

Perform additional validation by tracing SEND.

**Action**

1. Configure trace operations.

```
[edit protocols neighbor-discovery secure]
user@host# set traceoptions file send-log
user@host# set traceoptions flag all
```

2. Run the `show log` command.

```
user@host> show log send-log
Apr 11 06:21:26 proto: outgoing pkt on idx 68 does not have CGA (fe80::2a0:a514:0:14c), dropping pkt
Apr 11 06:26:44 proto: sendd_msg_handler: recv outgoing 96 bytes on idx 70 with offset 40
Apr 11 06:26:44 dbg: sendd_proto_handler: Modifier (16) 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Apr 11 06:26:44 cga: snd_is_lcl_cga: BEFORE overriding cc, cc:0, ws->col:0
Apr 11 06:26:44 proto: outgoing pkt on idx 70 does not have CGA
```
Meaning  The output shows that because the packet does not have a cryptographically generated address, the packet is dropped.

See Also  • Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 21
          • Understanding Secure IPv6 Neighbor Discovery on page 19
          • Understanding IPv6 Neighbor Discovery on page 46

Related Documentation  • Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
CHAPTER 6

Troubleshooting

- Working with Problems on Your Network on page 61
- Isolating a Broken Network Connection on page 62
- Identifying the Symptoms of a Broken Network Connection on page 63
- Isolating the Causes of a Network Problem on page 64
- Taking Appropriate Action for Resolving the Network Problem on page 65
- Evaluating the Solution to Check Whether the Network Problem Is Resolved on page 66

Working with Problems on Your Network

Problem Description: This checklist provides links to troubleshooting basics, an example network, and includes a summary of the commands you might use to diagnose problems with the router and network.

Table 3: Checklist for Working with Problems on Your Network

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Command or Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Isolating a Broken Network Connection” on page 62</td>
<td>ping (ip-address</td>
</tr>
<tr>
<td>1. Identifying the Symptoms of a Broken Network Connection on page 63</td>
<td>show &lt; configuration</td>
</tr>
<tr>
<td>2. Isolating the Causes of a Network Problem on page 64</td>
<td>[edit] delete routing options static route destination-prefix commit and-quit show route destination-prefix</td>
</tr>
<tr>
<td>3. Taking Appropriate Action for Resolving the Network Problem on page 65</td>
<td>show route (ip-address</td>
</tr>
<tr>
<td>4. Evaluating the Solution to Check Whether the Network Problem Is Resolved on page 66</td>
<td></td>
</tr>
</tbody>
</table>
Isolating a Broken Network Connection

By applying the standard four-step process illustrated in Figure 3 on page 62, you can isolate a failed node in the network. Note that the functionality described in this section is not supported in versions 15.1X49, 15.1X49-D30, or 15.1X49-D40.

**Figure 3: Process for Diagnosing Problems in Your Network**

Before you embark on the four-step process, however, it is important that you are prepared for the inevitable problems that occur on all networks. While you might find a solution to a problem by simply trying a variety of actions, you can reach an appropriate solution more quickly if you are systematic in your approach to the maintenance and monitoring of your network. To prepare for problems on your network, understand how the network functions under normal conditions, have records of baseline network activity, and carefully observe the behavior of your network during a problem situation.

**Figure 4 on page 62** shows the network topology used in this topic to illustrate the process of diagnosing problems in a network.

**Figure 4: Network with a Problem**

The network in Figure 4 on page 62 consists of two autonomous systems (ASs). AS 65001 includes two routers, and AS 65002 includes three routers. The border router (R1) in AS 65001 announces aggregated prefixes 100.100/24 to the AS 65002 network.
problem in this network is that R6 does not have access to R5 because of a loop between R2 and R6.

To isolate a failed connection in your network, follow the steps in these topics:

- **Identifying the Symptoms of a Broken Network Connection** on page 63
- **Isolating the Causes of a Network Problem** on page 64
- **Taking Appropriate Action for Resolving the Network Problem** on page 65
- **Taking Appropriate Action for Resolving the Network Problem** on page 65
- **Evaluating the Solution to Check Whether the Network Problem Is Resolved** on page 66

**Identifying the Symptoms of a Broken Network Connection**

**Problem Description:** The symptoms of a problem in your network are usually quite obvious, such as the failure to reach a remote host.

**Solution** To identify the symptoms of a problem on your network, start at one end of your network and follow the routes to the other end, entering all or one of the following Junos OS command-line interfaces (CLI) operational mode commands:

```
user@host> ping (ip-address | host-name)
user@host> show route (ip-address | host-name)
user@host> traceroute (ip-address | host-name)
```

**Sample Output**

```
user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
  4  5  00 0054 e2db 0000 01 01 a8c6 10.1.26.2 10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
  4  5  00 0054 e2de 0000 01 01 a8c3 10.1.26.2 10.0.0.5

36 bytes from 10.1.26.1: Time to live exceeded
Vr HL TOS Len ID Flg off TTL Pro cks Src Dst
  4  5  00 0054 e2e2 0000 01 01 a8bf 10.1.26.2 10.0.0.5

^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 0 packets received, 100% packet loss

user@R6> show route 10.0.0.5
inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.5/32  *[IS-IS/165] 00:02:39, metric 10
> to 10.1.26.1 via so-0/0/2.0
```
user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
1  10.1.26.1 (10.1.26.1)  0.649 ms  0.521 ms  0.490 ms
2  10.1.26.2 (10.1.26.2)  0.521 ms  0.537 ms  0.507 ms
3  10.1.26.1 (10.1.26.1)  0.523 ms  0.536 ms  0.514 ms
4  10.1.26.2 (10.1.26.2)  0.528 ms  0.551 ms  0.523 ms
5  10.1.26.1 (10.1.26.1)  0.531 ms  0.550 ms  0.524 ms

Meaning

The sample output shows an unsuccessful ping command in which the packets are being rejected because the time to live is exceeded. The output for the show route command shows the interface (10.1.26.1) that you can examine further for possible problems. The traceroute command shows the loop between 10.1.26.1 (R2) and 10.1.26.2 (R6), as indicated by the continuous repetition of the two interface addresses.

Isolating the Causes of a Network Problem

Problem Description: A particular symptom can be the result of one or more causes. Narrow down the focus of your search to find each individual cause of the unwanted behavior.

Solution

To isolate the cause of a particular problem, enter one or all of the following Junos OS CLI operational mode command:

```
user@host> show < configuration | bgp | interfaces | isis | ospf | route >
```

Your particular problem may require the use of more than just the commands listed above. See the appropriate command reference for a more exhaustive list of commonly used operational mode commands.

Sample Output

```
user@R6> show interfaces terse
Interface          Admin Link Proto Local                 Remote
so-0/0/0            up    up    inet  10.1.56.2/30
so-0/0/0.0          up    up    inet  10.1.56.2/30
so-0/0/2            up    up    inet  10.1.26.2/30
so-0/0/2.0          up    up    inet  10.1.26.2/30
so-0/0/3            up    up    inet  10.1.36.2/30
so-0/0/3.0          up    up    inet  10.1.36.2/30
[...Output truncated...]
```

The following sample output is from R2:

```
user@R2> show route 10.0.0.5
inet.0: 22 destinations, 25 routes (22 active, 0 holddown, 0 hidden)
```
Meaning

The sample output shows that all interfaces on R6 are up. The output from R2 shows that a static route [Static/5] configured on R2 points to R6 (10.1.26.2) and is the preferred route to R5 because of its low preference value. However, the route is looping from R2 to R6, as indicated by the missing reference to R5 (10.1.12.1).

Taking Appropriate Action for Resolving the Network Problem

**Problem Description:** The appropriate action depends on the type of problem you have isolated. In this example, a static route configured on R2 is deleted from the [routing-options] hierarchy level. Other appropriate actions might include the following:

- Check the local router’s configuration and edit it if appropriate.
- Troubleshoot the intermediate router.
- Check the remote host configuration and edit it if appropriate.
- Troubleshoot routing protocols.
- Identify additional possible causes.

To resolve the problem in this example, enter the following Junos OS CLI commands:

```plaintext
[edit]
user@R2# delete routing-options static route destination-prefix
user@R2# commit and-quit
user@R2# show route destination-prefix
```

**Sample Output**

```plaintext
[edit]
user@R2# delete routing-options static route 10.0.0.5/32

[edit]
user@R2# commit and-quit
commit complete
Exiting configuration mode

user@R2> show route 10.0.0.5
inet.0: 22 destinations, 24 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```
Meaning

The sample output shows the static route deleted from the [routing-options] hierarchy and the new configuration committed. The output for the show route command now shows the BGP route as the preferred route, as indicated by the asterisk (*).

Evaluating the Solution to Check Whether the Network Problem Is Resolved

Problem Description: If the problem is solved, you are finished. If the problem remains or a new problem is identified, start the process over again.

You can address possible causes in any order. In relation to the network in “Isolating a Broken Network Connection” on page 62, we chose to work from the local router toward the remote router, but you might start at a different point, particularly if you have reason to believe that the problem is related to a known issue, such as a recent change in configuration.

Solution

To evaluate the solution, enter the following Junos OS CLI commands:

```
user@host> show route (ip-address | host-name)
user@host> ping (ip-address | host-name)
user@host> traceroute (ip-address | host-name)
```

Sample Output

```
user@R6> show route 10.0.0.5
inet.0: 20 destinations, 20 routes (20 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.0.0.5/32 *[BGP/170] 00:01:35, MED 5, localpref 100
    AS path: 65001 I
    > to 10.1.26.1 via so-0/0/2.0

user@R6> ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
64 bytes from 10.0.0.5: icmp_seq=0 ttl=253 time=0.866 ms
64 bytes from 10.0.0.5: icmp_seq=1 ttl=253 time=0.837 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=253 time=0.796 ms
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.796/0.833/0.866/0.029 ms

user@R6> traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 40 byte packets
  1  10.1.26.1 (10.1.26.1)  0.629 ms  0.538 ms  0.497 ms
```
The sample output shows that there is now a connection between R6 and R5. The `show route` command shows that the BGP route to R5 is preferred, as indicated by the asterisk (*). The `ping` command is successful and the `traceroute` command shows that the path from R6 to R5 is through R2 (10.1.26.1), and then through R1 (10.1.12.1).
CHAPTER 7

Configuration Statements

- autonomous on page 70
- cryptographic-address on page 71
- current-hop-limit on page 72
- default-lifetime on page 73
- interface (Protocols IPv6 Neighbor Discovery) on page 74
- key-length on page 75
- key-pair on page 76
- link-mtu on page 77
- managed-configuration on page 78
- max-advertisement-interval (Protocols IPv6 Neighbor Discovery) on page 79
- min-advertisement-interval (Protocols IPv6 Neighbor Discovery) on page 80
- nd-retransmit-timer on page 81
- nd-system-cache-limit on page 82
- nd6-max-cache on page 83
- nd6-new-hold-limit on page 84
- neighbor-discovery on page 85
- on-link on page 86
- onlink-subnet-only on page 87
- other-stateful-configuration on page 88
- preference (IPv6 Router Advertisement) on page 89
- preferred-lifetime on page 90
- prefix (Protocols IPv6 Neighbor Discovery) on page 91
- reachable-time on page 92
- retransmit-timer on page 93
- router-advertisement on page 93
- secure on page 94
- security-level on page 95
- solicit-router-advertisement-unicast on page 95
### autonomous

**Syntax**

(autonomous | no-autonomous);

**Hierarchy Level**

[edit logical-systems logical-system-name protocols router-advertisement interface interface-name prefix prefix],
[edit protocols router-advertisement interface interface-name prefix prefix]

**Release Information**

Statement introduced before Junos OS Release 7.4.

**Description**

Specify whether prefixes in the router advertisement messages are used for stateless address autoconfiguration:

- **autonomous**—Use prefixes for address autoconfiguration.
- **no-autonomous**—Do not use prefixes for address autoconfiguration.

**Default**

autonomous

**Required Privilege**

routing—To view this statement in the configuration.

routinglevel—To add this statement to the configuration.

**Related Documentation**

- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
cryptographic-address

Syntax

cryptographic-address { key-length number; key-pair pathname; }

Hierarchy Level

[edit protocols neighbor-discovery secure]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure parameters for cryptographically generated addresses for Secure Neighbor Discovery.

The Secure Neighbor Discovery (SEND) Protocol uses cryptographically generated addresses (CGAs), as defined in RFC 3972, Cryptographically Generated Addresses, to ensure that the sender of a Neighbor Discovery Protocol (NDP) message is the “owner” of the claimed address. Each node must generate a public-private key pair before it can claim an address. The CGA is included in all outgoing neighbor solicitation and neighbor advertisement messages.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege

Level

routinglevel—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation

• Example: Configuring Secure IPv6 Neighbor Discovery on page 30
• Understanding Secure IPv6 Neighbor Discovery on page 19
current-hop-limit

**Syntax**

```
current-hop-limit number;
```

**Hierarchy Level**

```
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]
```

**Release Information**

Statement introduced before Junos OS Release 7.4.

**Description**

Set the default value placed in the hop count field of the IP header for outgoing packets.

**Options**

- **number**—Hop limit. A value of 0 means the limit is unspecified by this router.
  
  **Range:** 0 through 255
  
  **Default:** 64

**Required Privilege Level**

- **routing**—To view this statement in the configuration.
  
  **routing-control**—To add this statement to the configuration.

**Related Documentation**

- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
default-lifetime

**Syntax**
default-lifetime seconds;

**Hierarchy Level**
```
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]
```

**Release Information**
Statement introduced before Junos OS Release 7.4.

**Description**
Configure the lifetime associated with a default router.

**Options**
- **seconds**—Default lifetime. A value of 0 means this router is not the default router.
  - **Range:** Maximum advertisement interval value through 9000 seconds
  - **Default:** Three times the maximum advertisement interval value

**Required Privilege Level**
- routing—to view this statement in the configuration.
- routing-control—to add this statement to the configuration.

**Related Documentation**
- max-advertisement-interval on page 79
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
# interface (Protocols IPv6 Neighbor Discovery)

## Syntax

```
interface interface-name {
    current-hop-limit number;
    default-lifetime seconds;
    (link-mtu | no-link-mtu);
    (managed-configuration | no-managed-configuration);
    max-advertisement-interval seconds;
    min-advertisement-interval seconds;
    (other-stateful-configuration | no-other-stateful-configuration);
    prefix prefix {
        (autonomous | no-autonomous);
        (on-link | no-on-link);
        preferred-lifetime seconds;
        valid-lifetime seconds;
    }
    reachable-time milliseconds;
    retransmit-timer milliseconds;
    solicit-router-advertisement-unicast;
}
```

## Hierarchy Level

```
[edit logical-systems logical-system-name protocols router-advertisement],
[edit protocols router-advertisement]
```

## Release Information

Statement introduced before Junos OS Release 7.4. `solicit-router-advertisement-unicast` statement added from 15.1 Release onwards.

## Description

Configure router advertisement properties on an interface. To configure more than one interface, include the `interface` statement multiple times.

The Junos OS enters the Neighbor Discovery Protocol (NDP) packets into the routing platform cache even if there is no known route to the source.

If you are using Virtual Router Redundancy Protocol (VRRP) for IPv6, you must include the `virtual-router-only` statement on both the master and backup VRRP on the IPv6 router.

## Options

- `interface-name`—Name of an interface. Specify the full interface name, including the physical and logical address components.

  The remaining statements are explained separately. See CLI Explorer.

## Required Privilege Level

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

## Related Documentation

- [Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery](#)
- [Example: Configuring Secure IPv6 Neighbor Discovery](#)
## key-length

<table>
<thead>
<tr>
<th>Syntax</th>
<th>key-length number;</th>
</tr>
</thead>
</table>

### Hierarchy Level

[edit protocols neighbor-discovery secure cryptographic-address]

### Release Information

Statement introduced in Junos OS Release 9.3.

### Description

Specify the length of the RSA key used to generate the public-private key pair for the cryptographic address.

### Default

1024

### Options

- **number**—RSA key length.
  - **Range:** 1024 through 2048

### Required Privilege Level

- **routing**—To view this statement in the configuration.
- **routing-control**—To add this statement to the configuration.

### Related Documentation

- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
### key-pair

<table>
<thead>
<tr>
<th>Syntax</th>
<th>key-pair pathname;</th>
</tr>
</thead>
</table>

#### Hierarchy Level

[edit protocols neighbor-discovery secure cryptographic-address]

#### Release Information

Statement introduced in Junos OS Release 9.3.

#### Description

Specify the directory path of the public-private key file generated for the cryptographic address.

A cryptographic address is dynamically generated based on a public key and a subnet prefix.

#### Default

The default location of the file is the `/var/etc/rsa_key` directory.

#### Options

- `pathname`—Directory path of the public-private key file.

#### Required Privilege Level

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

#### Related Documentation

- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
**link-mtu**

Syntax  
(link-mtu | no-link-mtu);

Hierarchy Level  
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]

Release Information  
Statement introduced in Junos OS 10.3.

Description  
Specify whether to include the maximum transmission unit (MTU) option in router advertisement messages:

- **link-mtu**—Includes the MTU option in router advertisements.
- **no-link-mtu**—Does not include the MTU option in router advertisements.

The MTU option included in router advertisement messages ensures that all nodes on a link use the same MTU value in situations where the link MTU is not well known.

Default  
Router advertisement messages do not include the MTU option.

Required Privilege Level  
routing—To view this statement in the configuration.
router-control—To add this statement to the configuration.

Related Documentation  
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
### managed-configuration

| Syntax          | (managed-configuration | no-managed-configuration); |
|-----------------|-------------------------|
| Hierarchy Level | [edit logical-systems logical-system-name protocols router-advertisement interface interface-name], [edit protocols router-advertisement interface interface-name] |
| Release Information | Statement introduced before Junos OS Release 7.4. |
| Description | Specify whether to enable the host to use a stateful autoconfiguration protocol for address autoconfiguration, along with any stateless autoconfiguration already configured:

- **managed-configuration**—Enable host to use stateful autoconfiguration.
- **no-managed-configuration**—Disable host from using stateful autoconfiguration.

You can set two fields in the router advertisement message to enable stateful autoconfiguration on a host: the managed configuration field and the other stateful configuration field. Setting the managed configuration field enables the host to use a stateful autoconfiguration protocol for address autoconfiguration, along with any stateless autoconfiguration already configured. Setting the other stateful configuration field enables autoconfiguration of other nonaddress-related information.

| Default | Stateful autoconfiguration is disabled. |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | • Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46 • other-stateful-configuration on page 88 |
max-advertisement-interval (Protocols IPv6 Neighbor Discovery)

Syntax

```bash
max-advertisement-interval seconds;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Set the maximum interval between each router advertisement message.

The router sends router advertisements on each interface configured to transmit messages. The advertisements include route information and indicate to network hosts that the router is operational. The router sends these messages periodically, with a time range defined by minimum and maximum values.

Options

- `seconds`—Maximum interval.
  - Range: 4 through 1800 seconds
  - Default: 600 seconds

Required Privilege

- `routing`—To view this statement in the configuration.
- `routing-control`—To add this statement to the configuration.

Related Documentation

- [min-advertisement-interval on page 80](#)
- [Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46](#)
**min-advertisement-interval (Protocols IPv6 Neighbor Discovery)**

**Syntax**
```
min-advertisement-interval seconds;
```

**Hierarchy Level**
```
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]
```

**Release Information**
Statement introduced before Junos OS Release 7.4.

**Description**
Set the minimum interval between each router advertisement message.

The router sends router advertisements on each interface configured to transmit messages. The advertisements include route information and indicate to network hosts that the router is operational. The router sends these messages periodically, with a time range defined by minimum and maximum values.

**Options**
- `seconds`—Minimum interval.
  - **Range:** 3 seconds through three-quarter times the maximum advertisement interval value
  - **Default:** One-third the maximum advertisement interval value

By default, the maximum advertisement interval is 600 seconds and the minimum advertisement interval is one-third the maximum interval, or 200 seconds.

**Required Privilege Level**
- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**
- max-advertisement-interval on page 79
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
### nd-retransmit-timer

**Syntax**
```
nd-retransmit-timer milliseconds;
```

**Hierarchy Level**
[edit system]

**Release Information**
Statement introduced in Junos OS Release 15.1.

**Description**
Set the retransmit timer for neighbor discovery messages. Whenever the state of a neighbor during the Neighbor Discovery (ND) process changes from stale to probe, the value of the retransmit timer controls the interval between the neighbor solicitation messages that are sent out. Also, the retransmit timer controls the time for which the neighbor is in the probe state. A device sends a neighbor solicitation message after the specified number of milliseconds in the nd-retransmit-timer statement, until a reachability confirmation is received. If a solicited neighbor advertisement (NA) message is not received from the neighbor in response to the solicitation message sent from the device, the neighbor remains in the probe state.

**Options**
- **milliseconds**—Retransmission frequency.
- **Default:** 0 milliseconds

**Required Privilege Level**
- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**
- [Using NDRA to Provide IPv6 WAN Link Addressing Overview](#)

Copyright © 2019, Juniper Networks, Inc.
**nd-system-cache-limit**

**Syntax**
```plaintext
nd-system-cache-limit number;
```

**Hierarchy Level**
```
[edit system]
```

**Release Information**
Statement introduced in Junos OS Release 15.1.

**Description**
Specify the maximum system cache size for IPv6 next-hop addresses. This limit enforces a systemwide cap on the neighbor discovery cache entries for all interfaces, including the loopback interface for the internal routing instance, management interfaces, and the public interfaces.

**Default**
100,000

**Options**
- `number`—Maximum system cache size for IPv6 next-hop addresses.
  - **Range:** 200 through 2,000,000

**Required Privilege Level**
- admin—To view this statement in the configuration.
- admin-control—To add this statement to the configuration.

**Related Documentation**
- Example: Configuring Neighbor Discovery Cache Protection to Prevent Denial-of-Service Attacks on page 41
- `nd6-max-cache` on page 83
- `nd6-new-hold-limit` on page 84
### nd6-max-cache

**Syntax**
```
nd6-max-cache nd6-max-cache;
```

**Hierarchy Level**
```
[edit interfaces interface-name unit logical-unit-number family inet6]
```

**Release Information**
Statement introduced in Junos OS Release 15.1.

**Description**
Specify the maximum number of entries that can be added to the Neighbor Discovery Protocol (NDP) IPv6 neighbor discovery cache for an interface. When this maximum is reached, no new entries are allowed.

**Default**
- 100,000 for M Series.
- 75,000 for MX Series and QFX Series.
- 20,000 for EX Series.

**Options**
`nd6-max-cache`—Maximum size of the neighbor discovery next-hop cache for an interface.

- **Range**: 1 through 2,000,000 for MX Series or QFX Series.
- **Range**: 1 through 700,000 for M Series.

**Required Privilege Level**
- `interface`—To view this statement in the configuration.
- `interface-control`—To add this statement to the configuration.

**Related Documentation**
- Configuring Neighbor Discovery Cache Protection on page 40
- Example: Configuring Neighbor Discovery Cache Protection to Prevent Denial-of-Service Attacks on page 41
- IPv6 Neighbor Discovery Overview on page 17
- nd6-new-hold-limit on page 84
- nd-system-cache-limit on page 82
**nd6-new-hold-limit**

**Syntax**

```
nd6-new-hold-limit nd6-new-hold-limit;
```

**Hierarchy Level**

```
[edit interfaces interface-name unit logical-unit-number family inet6]
```

**Release Information**

Statement introduced in Junos OS Release 15.1.

**Description**

Specify the maximum number of entries for unresolved next-hop addresses that can be added to the Neighbor Discovery Protocol (NDP) IPv6 neighbor discovery cache for an interface.

**Default**

- 100,000 for M Series.
- 75,000 for MX Series and QFX Series.
- 20,000 for EX Series.

**Options**

`nd6-new-hold-limit`—Maximum number of new unresolved next-hop addresses that can be added to the IPv6 neighbor discovery cache.

**Range:** 1 through 2,000,000

**Required Privilege Level**

- interface—To view this statement in the configuration.
- interface-control—To add this statement to the configuration.

**Related Documentation**

- Configuring Neighbor Discovery Cache Protection on page 40
- Example: Configuring Neighbor Discovery Cache Protection to Prevent Denial-of-Service Attacks on page 41
- IPv6 Neighbor Discovery Overview on page 17
- `nd-system-cache-limit` on page 82
- `nd6-max-cache` on page 83
- `nd6-stale-time`
# neighbor-discovery

## Syntax

```
neighbor-discovery {
  onlink-subnet-only;
  secure {
    security-level {
      (default | secure-messages-only);
    }
    cryptographic-address {
      key-length number;
      key-pair pathname;
    }
    timestamp {
      clock-drift number;
      known-peer-window number;
      new-peer-window number;
    }
    traceoptions {
      file filename <files number> <match regular-expression> <size size> <world-readable | no-world-readable>;
      flag flag;
      no-remote-trace;
    }
  }
}
```

## Hierarchy Level

```
[edit protocols]
```

## Release Information

Statement introduced in Junos OS Release 9.3.

## Description

Enable Secure Neighbor Discovery.

The remaining statements are explained separately. See CLI Explorer.

## Default

Disabled

## Required Privilege Level

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

## Related Documentation

- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
## on-link

### Syntax

```plaintext
(on-link | no-on-link);
```

### Hierarchy Level

```plaintext
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name prefix prefix],
[edit protocols router-advertisement interface interface-name prefix prefix]
```

### Release Information

Statement introduced before Junos OS Release 7.4.

### Description

Specify whether to enable prefixes to be used for onlink determination:

- **no-on-link**—Disable prefixes from being used for onlink determination.
- **on-link**—Enable prefixes to be used for onlink determination.

Router advertisement messages carry prefixes and information about them. A prefix is onlink when it is assigned to an interface on a specified link. The prefixes specify whether they are onlink or not onlink. A node considers a prefix to be onlink if it is represented by one of the link’s prefixes, a neighboring router specifies the address as the target of a redirect message, a neighbor advertisement message is received for the (target) address, or any neighbor discovery message is received from the address. These prefixes are also used for address autoconfiguration. The information about the prefixes specifies the lifetime of the prefixes, whether the prefix is autonomous, and whether the prefix is onlink.

### Default

Prefixes are onlink unless explicitly disabled.

### Required Privilege

- **Level**
  - routing—To view this statement in the configuration.
  - routing-control—To add this statement to the configuration.

### Related Documentation

- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
**onlink-subnet-only**

**Syntax**

```
onlink-subnet-only;
```

**Hierarchy Level**

```
[edit logical-systems logical-system-name protocols neighbor-discovery],
[edit protocols neighbor-discovery]
```

**Release Information**

Statement introduced in Junos OS Release 10.0.
Statement introduced in Junos OS Release 11.3 for SRX Series devices.

**Description**

Enable this option to prevent the device from responding to a neighbor solicitation (NS) from a prefix that is not included as one of the device interface prefixes.

After configuring the `onlink-subnet-only` statement, the Routing Engine needs to be restarted using the `request system reboot both-routing-engines` command. If the attacker’s IPv6 destination address is already in the forwarding-table, it is not removed after you configure the `onlink-subnet-only` statement, and therefore the device continues to respond to ping NSs. Restarting the Routing Engine removes the entry from the forwarding table.

**Default**

Disabled

**Required Privilege**

```
admin— To view this statement in the configuration.
admin-control— To add this statement to the configuration.
```

**Related Documentation**

- **Understanding How to Control Inbound Traffic Based on Protocols**
- **IPv6 Neighbor Discovery Feature Guide**
other-stateful-configuration

Syntax
(\(\text{other-stateful-configuration} \mid \text{no-other-stateful-configuration}\));

Hierarchy Level
[edit logical-systems \(\text{logical-system-name}\) protocols router-advertisement \(\text{interface interface-name}\)],
[edit protocols router-advertisement \(\text{interface interface-name}\)]

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Specify whether to enable autoconfiguration of other nonaddress-related information:

- no-other-stateful-configuration—Disable autoconfiguration of other nonaddress-related information.
- other-stateful-configuration—Enable autoconfiguration of other nonaddress-related information.

Default
By default, stateful autoconfiguration is disabled.

Required Privilege Level
routing—To view this statement in the configuration.
 routing-control—To add this statement to the configuration.

Related Documentation
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
### preference (IPv6 Router Advertisement)

**Syntax**

```
preference (high | low | medium);
```

**Hierarchy Level**

```
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]
```

**Release Information**

Statement introduced in Junos OS Release 16.1 for the MX Series.

**Description**

Specify the router preference that is communicated to IPv6 hosts through router advertisements. The `preference` value in the router advertisements enables IPv6 hosts to select a default router to reach a remote destination.

The `preference` can be configured when there are multiple devices that route to distinct sets of prefixes and where one of the devices would lead to considerably fewer redirects. You can indicate a lower preference for a new device that is not completely configured yet, so that hosts do not adopt this new device as the default device and thus avoid traffic loss.

**Options**

You can specify different levels of preference depending on your requirements:

- **high**—Specify a high preference for a device.
- **low**—Specify a low preference for a device.
- **medium**—Specify a medium preference for a device. This is the default preference.

**Required Privilege Level**

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**

- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
**preferred-lifetime**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>preferred-lifetime seconds;</th>
</tr>
</thead>
</table>

**Hierarchy Level**

```
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name prefix prefix],
[edit protocols router-advertisement interface interface-name prefix prefix]
```

**Release Information**

Statement introduced before Junos OS Release 7.4.

**Description**

Specify how long the prefix generated by stateless autoconfiguration remains preferred.

**Options**

- **seconds**—Preferred lifetime, in seconds. If you set the preferred lifetime to 0xffffffff, the lifetime is infinite. The preferred lifetime is never greater than the valid lifetime.

  - **Default:** 604,800 seconds

**Required Privilege Level**

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**

- valid-lifetime on page 101
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
prefix (Protocols IPv6 Neighbor Discovery)

Syntax

```plaintext
prefix prefix {
  (autonomous | no-autonomous);
  (on-link | no-on-link);
  preferred-lifetime seconds;
  valid-lifetime seconds;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure prefix properties in router advertisement messages.

Options

- `prefix`—Prefix name.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

Related Documentation

- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
### reachable-time

<table>
<thead>
<tr>
<th>Syntax</th>
<th>reachable-time milliseconds;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy Level</td>
<td>[edit logical-systems logical-system-name protocols router-advertisement interface interface-name], [edit protocols router-advertisement interface interface-name]</td>
</tr>
</tbody>
</table>

**Release Information**

Statement introduced before Junos OS Release 7.4.

**Description**

Set the length of time that a node considers a neighbor reachable until another reachability confirmation is received from that neighbor.

After receiving a reachability confirmation from a neighbor, a node considers that neighbor reachable for a certain amount of time without receiving another confirmation. This mechanism is used for neighbor unreachability detection, a mechanism for finding link failures to a target node.

**Options**

- **milliseconds**—Reachability time limit.
  - **Range**: 0 through 3,600,000 milliseconds
  - **Default**: 0 milliseconds

**Required Privilege Level**

- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**

- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
retransmit-timer

Syntax: retransmit-timer milliseconds;

Hierarchy Level: [edit logical-systems logical-system-name protocols router-advertisement interface interface-name],
[edit protocols router-advertisement interface interface-name]

Release Information: Statement introduced before Junos OS Release 7.4.

Description: Set the retransmission frequency of neighbor solicitation messages. This timer is used to detect when a neighbor has become unreachable and to resolve addresses.

Options: milliseconds—Retransmission frequency.
Default: 0 milliseconds

Required Privilege Level: routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation:
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56

router-advertisement

Syntax: router-advertisement {...}

Hierarchy Level: [edit logical-systems logical-system-name protocols],
[edit protocols]

Release Information: Statement introduced before Junos OS Release 7.4.

Description: Enable router advertisement.
The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level: routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation:
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
secure

Syntax

secure {
  security-level {
    (default | secure-messages-only);
  }
  cryptographic-address {
    key-length number;
    key-pair pathname;
  }
  timestamp {
    clock-drift number;
    known-peer-window seconds;
    new-peer-window seconds;
  }
  traceoptions {
    file filename <files number> <match regular-expression> <size size> <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
  }
}

Hierarchy Level

[edit protocols neighbor-discovery]

Release Information
Statement introduced in Junos OS Release 9.3.

Description
Configure parameters for Secure Neighbor Discovery.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation
• Example: Configuring Secure IPv6 Neighbor Discovery on page 56
security-level

Syntax

```
security-level {
  (default | secure-messages-only);
}
```

Hierarchy Level

```
[edit protocols neighbor-discovery secure]
```

Release Information
Statement introduced in Junos OS Release 9.3.

Description
Configure the type of security mode for Secure Neighbor Discovery.

Options
- **default**—Accept and transmit both secure and unsecured messages.
- **secure-messages-only**—Accept secure messages only. Discard unsecured messages.

Required Privilege Level
- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

Related Documentation
- Example: Configuring Secure IPv6 Neighbor Discovery on page 56

solicit-router-advertisement-unicast

Syntax

```
solicit-router-advertisement-unicast;
```

Hierarchy Level

```
[edit protocols router-advertisement interface interface-name]
```

Release Information
Statement introduced in Junos OS Release 15.1R1 onwards.

Description
Configure devices to send router advertisements as unicast in response to the router solicitation message sent by IPv6 routers.

Required Privilege Level
- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

Related Documentation
- IPv6 Neighbor Discovery Feature Guide
timestamp

Syntax

```plaintext
timestamp {
    clock-drift value;
    known-peer-window seconds;
    new-peer-window seconds;
}
```

Hierarchy Level

[edit protocols neighbor-discovery secure]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure timestamp options, which are used to ensure that solicitation and redirect messages are not being replayed.

Options

- **clock-drift value**—Specify the allowable drift in time between the synchronization of peers. For `value`, specify a fractional value of 100.
  
  **Default:** 0.01

- **known-peer-window seconds**—Specify the expected interval in seconds between Secure Neighbor Discovery messages from an established peer. A message from a known peer that arrives after the specified interval is discarded.
  
  **Default:** 1 second

- **new-peer-window seconds**—Specify the maximum allowable time in seconds between the timestamp of a Secure Neighbor Discovery message from a new peer and when it can be accepted.
  
  **Default:** 300 seconds

Required Privilege

- **routing**—To view this statement in the configuration.
- **routing-control**—To add this statement to the configuration.

Related Documentation

- Example: Configuring Secure IPv6 Neighbor Discovery on page 56
traceoptions (Protocols IPv6 Neighbor Discovery)

Syntax

```
traceoptions {
  file filename <files number> <size size> <world-readable | no-world-readable>;
  flag flag <disable>;
}
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols router-advertisement],
[edit protocols router-advertisement]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For IPv6 neighbor discovery, specify router advertisement protocol-level tracing options. Trace IPv6 Neighbor Discovery protocol traffic to help debug Neighbor Discovery protocol issues.

Global tracing options are inherited from the configuration set by the `traceoptions` statement at the [edit routing-options] hierarchy level. You can override the following global trace options for the IPv6 Neighbor Discovery protocol using the `traceoptions flag` statement included at the [edit protocols router-advertisement] hierarchy level:

Default

The default trace options are inherited from the global `traceoptions` statement.

Options

- `disable`—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as `all`.

- `file filename`—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place router advertisement tracing output in the file `/var/log/router-advertisement-log`.

- `files number`—(Optional) Maximum number of trace files. When a trace file named `trace-file` reaches its maximum size, it is renamed `trace-file.0`, then `trace-file.1`, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum number of files, you must also specify a maximum file size with the `size` option.

  Range: 2 through 1000 files

  Default: 10 files

- `flag flag`—Tracing operation to perform. To specify more than one tracing operation, include multiple `flag` statements.

  - `all`—All tracing operations
**NOTE:** Use the trace flag all with caution as this may cause the CPU to become very busy.

- **general**—A combination of the normal and route trace operations
- **normal**—All normal operations.

**Default:** If you do not specify this option, only unusual or abnormal operations are traced.

- **policy**—Policy operations and actions
- **route**—Routing table changes
- **state**—State transitions
- **task**—IPv6 interface transactions and processing
- **timer**—IPv6 neighbor discovery protocol timer processing

**no-world-readable**—(Optional) Prevent any user from reading the log file.

**size size**—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). When a trace file named `trace-file` reaches this size, it is renamed `trace-file.0`. When the `trace-file` again reaches its maximum size, `trace-file.0` is renamed `trace-file.1` and `trace-file` is renamed `trace-file.0`. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum file size, you must also specify a maximum number of trace files with the `files` option.

**Syntax:** `x k` to specify KB, `x m` to specify MB, or `x g` to specify GB

**Range:** 10 KB through the maximum file size supported on your system

**Default:** 128 KB

**world-readable**—(Optional) Allow any user to read the log file.

**Required Privilege Level**
- routing—To view this statement in the configuration.
- routing-control—To add this statement to the configuration.

**Related Documentation**
- Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
traceoptions (Protocols Secure Neighbor Discovery)

Syntax
```
traceoptions {
  file filename <files number> <match regular-expression> <size size> <world-readable |
  no-world-readable >;
  flag flag;
  no-remote-trace;
}
```

Hierarchy Level
```
[edit protocols neighbor-discovery secure]
```

Release Information
Statement introduced in Junos OS Release 9.3.

Description
Configure tracing operations for Secure Neighbor Discovery events. To specify more than one tracing operation, include multiple flag statements.

Options
- **file filename**—Name of the file to receive the tracing operation. Enclose the name within quotation marks. All files are placed in the directory `/var/log`.
- **files number**—(Optional) Maximum number of trace files. When a trace file named `trace-file` reaches its maximum size, it is renamed `trace-file.0`, then `trace-file.1` and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.
  - If you specify a maximum number of files, you must also specify a maximum file size with the size option.
  - **Range**: 2 through 1000 files
  - **Default**: 10 files
- **flag**—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements.
  - **Secure Neighbor Discovery Tracing Options**
    - **configuration**—All configuration events.
    - **cryptographic-address**—Cryptographically generated address events.
    - **protocol**—All protocol processing events.
    - **rsa**—RSA events.
  - **Global Tracing Options**
    - **all**—All tracing operations.

You can specify one or more of following flag modifiers:
- **detail**—Provide detailed trace information.
- **receive**—Packets being received.
• send—Packets being transmitted.

match regular-expression—(Optional) Specify a regular expression to match the output of the trace file you want to log.

no-remote-trace—Disable remote tracing globally or for a specific tracing operation.

no-world-readable—(Optional) Prevent any user from reading this log file.

size size—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). When a trace file named trace-file reaches this size, it is renamed trace-file.0. When the trace-file again reaches its maximum size, trace-file.0 is renamed trace-file.1, and trace-file is renamed trace-file.0. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

Syntax:  \xk to specify KB, \xm to specify MB, or \xg to specify GB

Range:  10 KB through the maximum file size supported on your system

Default:  128 KB

world-readable—(Optional) Allow any user to read this log file.

Required Privilege
Level
routing and trace—To view this statement in the configuration.
routing-control and trace-control—To add this statement to the configuration.

Related Documentation
• Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 21
• Understanding Secure IPv6 Neighbor Discovery on page 19
• Understanding IPv6 Neighbor Discovery on page 46
valid-lifetime

Syntax
valid-lifetime seconds;

Hierarchy Level
[edit logical-systems logical-system-name protocols router-advertisement interface interface-name prefix prefix],
[edit protocols router-advertisement interface interface-name prefix prefix]

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Specify how long the prefix remains valid for onlink determination.

Options
seconds—Valid lifetime, in seconds. If you set the valid lifetime to 0xffffffff, the lifetime is infinite.

Default: 2,592,000 seconds

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation
• preferred-lifetime on page 90
• Example: Configuring IPv6 Interfaces and Enabling Neighbor Discovery on page 46
CHAPTER 8

Operational Commands

- clear ipv6 neighbors
- clear ipv6 router-advertisement
- monitor interface
- monitor start
- monitor stop
- ping
- show ipv6 neighbors
- show ipv6 router-advertisement
- show log
- traceroute
## clear ipv6 neighbors

### Syntax

```
clear ipv6 neighbors
<all | host hostname>
```

### Release Information

- Command introduced before Junos OS Release 7.4.
- Command introduced in Junos OS Release 9.3 for EX Series switches.
- Command introduced in Junos OS Release 12.2 for the QFX Series.
- Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

### Description

Clear IPv6 neighbor cache information.

**NOTE:** On Junos OS Evolved, issuing the `clear ipv6 neighbors` command clears the cache for IPv6 neighbors in a reachable state.

### Options

- **none**—Clear all IPv6 neighbor cache information.
- **all**—(Optional) Clear all IPv6 neighbor cache information.
- **host hostname**—(Optional) Clear the information for the specified IPv6 neighbors.

### Required Privilege Level

- `view`

### Related Documentation

- [show ipv6 neighbors on page 128](#)

### List of Sample Output

- **clear ipv6 neighbors on page 104**

### Output Fields

When you enter this command, you are provided feedback on the status of your request.

### Sample Output

```
clear ipv6 neighbors

user@host> clear ipv6 neighbors
```
clear ipv6 router-advertisement

Syntax

<table>
<thead>
<tr>
<th>clear ipv6 router-advertisement</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface interface&gt;</td>
</tr>
<tr>
<td>&lt;logical-system (all</td>
</tr>
</tbody>
</table>

Release Information

Command introduced before Junos OS Release 7.4.

Description

Clear IPv6 router advertisement counters.

Options

- none—Clear IPv6 router advertisement counters for all interfaces.
- interface interface—(Optional) Clear IPv6 router advertisement counters for the specified interface.
- logical-system (all | logical-system-name)—(Optional) Perform this operation on all logical systems or on a particular logical system.

Required Privilege

view

Related Documentation

- show ipv6 router-advertisement on page 131

List of Sample Output

clear ipv6 router-advertisement on page 105

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear ipv6 router-advertisement

user@host> clear ipv6 router-advertisement
monitor interface

Syntax

```
monitor interface
  <interface-name> | traffic <detail>>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display real-time statistics about interfaces, updating the statistics every second. Check for and display common interface failures, such as SONET/SDH and T3 alarms, loopbacks detected, and increases in framing errors.

**NOTE:** On Junos OS Evolved, you can use the `monitor interface` command over SSH sessions, but console and Telnet sessions are not supported.

**NOTE:** This command is not supported on the QFX3000 QFabric switch.

Options

- **none**—Display real-time statistics for all interfaces.
- **detail**—(Optional) With traffic option only, display detailed output.
- **interface-name**—(Optional) Display real-time statistics for the specified interface. In a TX Matrix or TX Matrix Plus router, display real-time statistics for the physical interfaces on the specified line-card chassis (LCC) only.
- **traffic**—(Optional) Display traffic data for all active interfaces. In a TX Matrix or TX Matrix Plus router, display real-time statistics for the physical interfaces on the specified LCC only.

Additional Information

The output of this command shows how much each field has changed since you started the command or since you cleared the counters by pressing the c key. For a description of the statistical information provided in the output of this command, see the `show interfaces extensive` command for a particular interface type in the CLI Explorer. To control the output of the `monitor interface` command while it is running, use the keys listed in Table 4 on page 107. The keys are not case-sensitive.
Table 4: Output Control Keys for the monitor interface interface-name Command

<table>
<thead>
<tr>
<th>Key</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>Clears (returns to zero) the delta counters since monitor interface was started. This does not clear the accumulative counter. To clear the accumulative counter, use the clear interfaces interval command.</td>
</tr>
<tr>
<td>f</td>
<td>Freezes the display, halting the display of updated statistics and delta counters.</td>
</tr>
<tr>
<td>i</td>
<td>Displays information about a different interface. The command prompts you for the name of a specific interface.</td>
</tr>
<tr>
<td>n</td>
<td>Displays information about the next interface. The monitor interface command displays the physical or logical interfaces in the same order as the show interfaces terse command.</td>
</tr>
<tr>
<td>q or Esc</td>
<td>Quits the command and returns to the command prompt.</td>
</tr>
<tr>
<td>t</td>
<td>Thawsthe display, resuming the update of the statistics and delta counters.</td>
</tr>
</tbody>
</table>

To control the output of the monitor interface traffic command while it is running, use the keys listed in Table 5 on page 107. The keys are not case-sensitive.

Table 5: Output Control Keys for the monitor interface traffic Command

<table>
<thead>
<tr>
<th>Key</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>Displays the statistics in units of bytes and bytes per second (bps).</td>
</tr>
<tr>
<td>c</td>
<td>Clears (return to 0) the delta counters in the Current Delta column. The statistics counters are not cleared.</td>
</tr>
<tr>
<td>d</td>
<td>Displays the Current Delta column (instead of the rate column) in bps or packets per second (pps).</td>
</tr>
<tr>
<td>p</td>
<td>Displays the statistics in units of packets and packets per second (pps).</td>
</tr>
<tr>
<td>q or Esc</td>
<td>Quits the command and returns to the command prompt.</td>
</tr>
<tr>
<td>r</td>
<td>Displays the rate column (instead of the Current Delta column) in bps and pps.</td>
</tr>
</tbody>
</table>

Required Privilege Level: trace

List of Sample Output
- monitor interface (Physical) on page 109
- monitor interface (OTN Interface) on page 110
- monitor interface (MX480 Router with MPC5E and 10-Gigabit Ethernet OTN Interface) on page 111
- monitor interface (MX480 Router with MPC5E and 100-Gigabit Ethernet Interface) on page 112
Table 6 on page 108 describes the output fields for the `monitor interface` command. Output fields are listed in the approximate order in which they appear.

**Output Fields**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>router1</strong></td>
<td>Hostname of the router.</td>
<td>All levels</td>
</tr>
<tr>
<td><strong>Seconds</strong></td>
<td>How long the monitor interface command has been running or how long since you last cleared the counters.</td>
<td>All levels</td>
</tr>
<tr>
<td><strong>Time</strong></td>
<td>Current time (UTC).</td>
<td>All levels</td>
</tr>
<tr>
<td><strong>Delay x/y/z</strong></td>
<td>Time difference between when the statistics were displayed and the actual clock time. &lt;br&gt;• <em>x</em>—Time taken for the last polling (in milliseconds). &lt;br&gt;• <em>y</em>—Minimum time taken across all pollings (in milliseconds). &lt;br&gt;• <em>z</em>—Maximum time taken across all pollings (in milliseconds).</td>
<td>All levels</td>
</tr>
<tr>
<td><strong>Interface</strong></td>
<td>Short description of the interface, including its name, status, and encapsulation.</td>
<td>All levels</td>
</tr>
<tr>
<td><strong>Link</strong></td>
<td>State of the link: Up, Down, or Test.</td>
<td>All levels</td>
</tr>
<tr>
<td><strong>Current delta</strong></td>
<td>Cumulative number for the counter in question since the time shown in the Seconds field, which is the time since you started the command or last cleared the counters.</td>
<td>All levels</td>
</tr>
<tr>
<td><strong>Local Statistics</strong></td>
<td>(Logical interfaces only) Number and rate of bytes and packets destined to the router or switch through the specified interface. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It usually takes less than 1 second for this counter to stabilize. &lt;br&gt;• Input bytes—Number of bytes received on the interface. &lt;br&gt;• Output bytes—Number of bytes transmitted on the interface. &lt;br&gt;• Input packets—Number of packets received on the interface. &lt;br&gt;• Output packets—Number of packets transmitted on the interface.</td>
<td>All levels</td>
</tr>
</tbody>
</table>
Table 6: monitor interface Output Fields (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
<th>Level of Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Remote Statistics</strong></td>
<td>(Logical interfaces only) Statistics for traffic transiting the router or switch. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It usually takes less than 1 second for this counter to stabilize.</td>
<td>All levels</td>
</tr>
<tr>
<td>• Input bytes</td>
<td>Number of bytes received on the interface.</td>
<td></td>
</tr>
<tr>
<td>• Output bytes</td>
<td>Number of bytes transmitted on the interface.</td>
<td></td>
</tr>
<tr>
<td>• Input packets</td>
<td>Number of packets received on the interface.</td>
<td></td>
</tr>
<tr>
<td>• Output packets</td>
<td>Number of packets transmitted on the interface.</td>
<td></td>
</tr>
<tr>
<td><strong>Traffic statistics</strong></td>
<td>Total number of bytes and packets received and transmitted on the interface. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It usually takes less than 1 second for this counter to stabilize.</td>
<td>All levels</td>
</tr>
<tr>
<td>• Input bytes</td>
<td>Number of bytes received on the interface.</td>
<td></td>
</tr>
<tr>
<td>• Output bytes</td>
<td>Number of bytes transmitted on the interface.</td>
<td></td>
</tr>
<tr>
<td>• Input packets</td>
<td>Number of packets received on the interface.</td>
<td></td>
</tr>
<tr>
<td>• Output packets</td>
<td>Number of packets transmitted on the interface.</td>
<td></td>
</tr>
</tbody>
</table>

**Description**

With the `traffic` option, displays the interface description configured at the [edit interfaces interface-name] hierarchy level.

### Sample Output

**monitor interface (Physical)**

```
user@host> monitor interface so-0/0/0

router1  Seconds: 19  Time: 15:46:29

Interface: so-0/0/0, Enabled, Link is Up
Encapsulation: PPP, Keepalives, Speed: OC48
Traffic statistics:
  Input packets: 6045 (0 pps) [11]
  Input bytes: 6290065 (0 bps) [13882]
  Output packets: 10376 (0 pps) [10]
  Output bytes: 10365540 (0 bps) [9418]
Encapsulation statistics:
  Input keepalives: 1901 [2]
  Output keepalives: 1901 [2]
NCP state: Opened
LCP state: Opened
Error statistics:
  Input errors: 0 [0]
  Input drops: 0 [0]
  Input framing errors: 0 [0]
  Policed discards: 0 [0]
  L3 incompletes: 0 [0]
  L2 channel errors: 0 [0]
  L2 mismatch timeouts: 0 [0]
  Carrier transitions: 1 [0]
  Output errors: 0 [0]
```
monitor interface (OTN interface)

user@host> monitor interface ge-7/0/0

Interface: ge-7/0/0, Enabled, Link is Up
Encapsulation: Ethernet, Speed: 10000mbps
Traffic statistics:

- Input bytes: 0 (0 bps)
- Output bytes: 0 (0 bps)
- Input packets: 0 (0 pps)
- Output packets: 0 (0 pps)

Error statistics:

- Input errors: 0
- Input drops: 0
- Input framing errors: 0
- Policed discards: 0
- L3 incompletes: 0
- L2 channel errors: 0
- L2 mismatch timeouts: 0
- Carrier transitions: 5
- Output errors: 0
- Output drops: 0
- Aged packets: 0

Active alarms: None
Active defects: None

Input MAC/Filter statistics:

- Unicast packets: 0
- Broadcast packets: 0
- Multicast packets: 0
- Oversized frames: 0
Packet reject count                  0
DA rejects                           0
SA rejects                           0
Output MAC/Filter Statistics:
    Unicast packets                      0
    Broadcast packets                    0
    Multicast packets                    0
    Packet pad count                     0
    Packet error count                   0
OTN Link 0
    OTN Alarms: OTU_BDI, OTU_TTIM, ODU_BDI
    OTN Defects: OTU_BDI, OTU_TTIM, ODU_BDI, ODU_TTIM
OTN OC - Seconds
    LOS                                  2
    LOF                                  9
OTN OTU - FEC Statistics
    Corr err ratio                      N/A
    Corr bytes                          0
    Uncorr words                        0
OTN OTU - Counters
    BIP                                  0
    BBE                                  0
    ES                                   0
    SES                                  0
    UAS                                  422
OTN ODU - Counters
    BIP                                  0
    BBE                                  0
    ES                                   0
    SES                                  0
    UAS                                  422
OTN ODU - Received Overhead APSPCC 0-3:     0

monitor interface (MX480 Router with MPC5E and 10-Gigabit Ethernet OTN Interface)

user@host> monitor interface xe-0/0/3

Interface: xe-0/0/3, Enabled, Link is Up
Encapsulation: Ethernet, Speed: 10000mbps
Traffic statistics: Current delta
    Input bytes:                               0 ( 0 bps)              [ 0]
    Output bytes:                              0 ( 0 bps)              [ 0]
    Input packets:                             0 ( 0 pps)              [ 0]
    Output packets:                            0 ( 0 pps)              [ 0]
Error statistics:
    Input errors:                              0                  [ 0]
    Input drops:                                0                  [ 0]
    Input framing errors:                     0                  [ 0]
    Policed discards:                          0                  [ 0]
    L3 incompletes:                            0                  [ 0]
    L2 channel errors:                        0                  [ 0]
    L2 mismatch timeouts:                     0                  [ 0]
    Carrier transitions:                      5                  [ 0]
    Output errors:                             0                  [ 0]
    Output drops:                              0                  [ 0]
    Aged packets:                              0                  [ 0]
Active alarms : None
Active defects: None
PCS statistics: Seconds
    Bit Errors                             0                  [ 0]
Errored blocks       4

Input MAC/Filter statistics:
  Unicast packets       0
  Broadcast packets     0
  Multicast packets     0
  Oversized frames      0
  Packet reject count   0
  DA rejects            0
  SA rejects            0

Output MAC/Filter Statistics:
  Unicast packets       0
  Broadcast packets     0
  Multicast packets     0
  Packet pad count      0
  Packet error count    0

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'

Monitor interface (MX480 Router with MPC5E and 100-Gigabit Ethernet Interface)

user@host> monitor interface et-2/1/0

Interface: et-2/1/0, Enabled, Link is Up
Encapsulation: Ethernet, Speed: 100000mbps

Traffic statistics:                                                 Current delta
  Input bytes:                         0 (0 bps)                           [0]
  Output bytes:                        0 (0 bps)                           [0]
  Input packets:                       0 (0 pps)                           [0]
  Output packets:                      0 (0 pps)                           [0]

Error statistics:
  Input errors:                        0                                   [0]
  Input drops:                         0                                   [0]
  Input framing errors:                0                                   [0]
  Policed discards:                    0                                   [0]
  L3 incompletes:                      0                                   [0]
  L2 channel errors:                   0                                   [0]
  L2 mismatch timeouts:                0                                   [0]
  Carrier transitions:                263                                   [0]
  Output errors:                       0                                   [0]
  Output drops:                        0                                   [0]
  Aged packets:                        0                                   [0]

OTN Link 0
OTN Alarms:
  OTN OC - Seconds
    LOS                             129                                   [0]
    LOF                             2                                   [0]

OTN OTU - FEC Statistics
  Corr err ratio                   <8E-5
  Corr bytes                       169828399453                                   [0]
  Uncorr words                     28939961456                                   [0]

OTN OTU - Counters
  BIP                              0                                   [0]
  BBE                              0                                   [0]
  ES                               24                                   [0]
  SES                              0                                   [0]
  UAS                              1255                                   [0]

OTN ODU - Counters

Copyright © 2019, Juniper Networks, Inc.
<table>
<thead>
<tr>
<th>Metric</th>
<th>Current</th>
<th>Delta</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIP</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BBE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ES</td>
<td>24</td>
<td>0</td>
</tr>
<tr>
<td>SES</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UAS</td>
<td>1256</td>
<td>0</td>
</tr>
<tr>
<td>OTN ODU - Received Overhead</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>APSPCC 0-3:</td>
<td>00 00 00 00</td>
<td></td>
</tr>
</tbody>
</table>

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'

**monitor interface (MX2010 Router with MPC6E and 10-Gigabit Ethernet OTN Interface)**

```
user@host> monitor interface xe-6/1/0
```

**Interface:** xe-6/1/0, Enabled, Link is Up

**Encapsulation:** Ethernet, **Speed:** 10000mbps

**Traffic statistics:**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Current</th>
<th>Delta</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input bytes:</td>
<td>0 (0 bps)</td>
<td>0</td>
</tr>
<tr>
<td>Output bytes:</td>
<td>0 (0 bps)</td>
<td>0</td>
</tr>
<tr>
<td>Input packets:</td>
<td>0 (0 pps)</td>
<td>0</td>
</tr>
<tr>
<td>Output packets:</td>
<td>0 (0 pps)</td>
<td>0</td>
</tr>
</tbody>
</table>

**Error statistics:**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input errors:</td>
<td>0</td>
</tr>
<tr>
<td>Input drops:</td>
<td>0</td>
</tr>
<tr>
<td>Input framing errors:</td>
<td>0</td>
</tr>
<tr>
<td>Policed discards:</td>
<td>0</td>
</tr>
<tr>
<td>L3 incompletes:</td>
<td>0</td>
</tr>
<tr>
<td>L2 channel errors:</td>
<td>0</td>
</tr>
<tr>
<td>L2 mismatch timeouts:</td>
<td>0</td>
</tr>
<tr>
<td>Carrier transitions:</td>
<td>1</td>
</tr>
<tr>
<td>Output errors:</td>
<td>0</td>
</tr>
<tr>
<td>Output drops:</td>
<td>0</td>
</tr>
<tr>
<td>Aged packets:</td>
<td>0</td>
</tr>
</tbody>
</table>

**Active alarms:** None

**Active defects:** None

**PCS statistics:**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit Errors</td>
<td>0</td>
</tr>
<tr>
<td>Errored blocks</td>
<td>1</td>
</tr>
</tbody>
</table>

**Input MAC/Filter statistics:**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unicast packets</td>
<td>0</td>
</tr>
<tr>
<td>Broadcast packets</td>
<td>0</td>
</tr>
<tr>
<td>Multicast packets</td>
<td>0</td>
</tr>
<tr>
<td>Oversized frames</td>
<td>0</td>
</tr>
<tr>
<td>Packet reject count</td>
<td>0</td>
</tr>
<tr>
<td>DA rejects</td>
<td>0</td>
</tr>
<tr>
<td>SA rejects</td>
<td>0</td>
</tr>
</tbody>
</table>

**Output MAC/Filter Statistics:**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unicast packets</td>
<td>0</td>
</tr>
<tr>
<td>Broadcast packets</td>
<td>0</td>
</tr>
<tr>
<td>Multicast packets</td>
<td>0</td>
</tr>
<tr>
<td>Packet pad count</td>
<td>0</td>
</tr>
<tr>
<td>Packet error count</td>
<td>0</td>
</tr>
</tbody>
</table>

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'
monitor interface (MX2010 Router with MPC6E and 100-Gigabit Ethernet OTN Interface)

```
user@host>  monitor interface et-9/0/0
Interface: et-9/0/0, Enabled, Link is Up
Encapsulation: Ethernet, Speed: 100000mbps
Traffic statistics:                                                        Current delta
  Input bytes:                       0 (0 bps)                           [0]
  Output bytes:                      0 (0 bps)                           [0]
  Input packets:                     0 (0 pps)                           [0]
  Output packets:                    0 (0 pps)                           [0]
Error statistics:
  Input errors:                      0                                   [0]
  Input drops:                       0                                   [0]
  Input framing errors:              0                                   [0]
  Policed discards:                  0                                   [0]
  L3 incompletes:                    0                                   [0]
  L2 channel errors:                 0                                   [0]
  L2 mismatch timeouts:              0                                   [0]
  Carrier transitions:               1                                   [0]
  Output errors:                     0                                   [0]
  Output drops:                      0                                   [0]
  Aged packets:                      0                                   [0]

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'
```

monitor interface (MX2020 Router with MPC6E and 10-Gigabit Ethernet OTN Interface)

```
user@host>  monitor interface xe-3/0/0
host name                           Seconds: 67                  Time: 23:46:46
Delay: 0/0/13                             
Interface: xe-3/0/0, Enabled, Link is Up
Encapsulation: Ethernet, Speed: 10000mbps
Traffic statistics:                                                        Current delta
  Input bytes:                       0 (0 bps)                           [0]
  Output bytes:                      0 (0 bps)                           [0]
  Input packets:                     0 (0 pps)                           [0]
  Output packets:                    0 (0 pps)                           [0]
Error statistics:
  Input errors:                      0                                   [0]
  Input drops:                       0                                   [0]
  Input framing errors:              0                                   [0]
  Policed discards:                  0                                   [0]
  L3 incompletes:                    0                                   [0]
  L2 channel errors:                 0                                   [0]
  L2 mismatch timeouts:              0                                   [0]
  Carrier transitions:               3                                   [0]
  Output errors:                     0                                   [0]
  Output drops:                      0                                   [0]
  Aged packets:                      0                                   [0]

OTN Link 0
OTN Alarms:
OTN Defects:
OTN OC - Seconds
  LOS                                0                                   [0]
  LOF                                0                                   [0]
```
### OTN OTU - FEC Statistics

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Corr err ratio</td>
<td>N/A</td>
<td>[0]</td>
</tr>
<tr>
<td>Corr bytes</td>
<td>0</td>
<td>[0]</td>
</tr>
<tr>
<td>Uncorr words</td>
<td>0</td>
<td>[0]</td>
</tr>
</tbody>
</table>

### OTN OTU - Counters

<table>
<thead>
<tr>
<th>Counter</th>
<th>Value</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>BIP</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>BBE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>ES</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SES</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>UAS</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### OTN ODU - Counters

<table>
<thead>
<tr>
<th>Counter</th>
<th>Value</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>BIP</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>BBE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>ES</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SES</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>UAS</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

### OTN ODU - Received Overhead

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>APSPCC 0-3</td>
<td>00 00 00 00</td>
<td></td>
</tr>
</tbody>
</table>

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'

---

**monitor interface (Logical)**

```shell
terminology shell> monitor interface so-1/0/0.0
```

host name | Seconds: 16 | Time: 15:33:39 | Delay: 0/0/1
----------|-------------|----------------|----------------

Interface: so-1/0/0.0, Enabled, Link is Down
Flags: Hardware-Down Point-To-Point SNMP-Traps

Local statistics:

<table>
<thead>
<tr>
<th></th>
<th>Current delta</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input bytes</td>
<td>0</td>
</tr>
<tr>
<td>Output bytes</td>
<td>0</td>
</tr>
<tr>
<td>Input packets</td>
<td>0</td>
</tr>
<tr>
<td>Output packets</td>
<td>0</td>
</tr>
</tbody>
</table>

Remote statistics:

<table>
<thead>
<tr>
<th></th>
<th>Current delta</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input bytes</td>
<td>0 (0 bps)</td>
</tr>
<tr>
<td>Output bytes</td>
<td>0 (0 bps)</td>
</tr>
<tr>
<td>Input packets</td>
<td>0 (0 pps)</td>
</tr>
<tr>
<td>Output packets</td>
<td>0 (0 pps)</td>
</tr>
</tbody>
</table>

Traffic statistics:

Destination address: 192.168.8.193, Local: 192.168.8.21

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'

---

**monitor interface (QFX3500 Switch)**

```shell
terminology shell> monitor interface ge-0/0/0
```

Interface: ge-0/0/0, Enabled, Link is Down
Encapsulation: Ethernet, Speed: Unspecified

Traffic statistics:

<table>
<thead>
<tr>
<th></th>
<th>Current delta</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input bytes</td>
<td>0 (0 bps)</td>
</tr>
<tr>
<td>Output bytes</td>
<td>0 (0 bps)</td>
</tr>
<tr>
<td>Input packets</td>
<td>0 (0 pps)</td>
</tr>
<tr>
<td>Output packets</td>
<td>0 (0 pps)</td>
</tr>
</tbody>
</table>

Error statistics:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Input errors</td>
<td>0</td>
</tr>
</tbody>
</table>

Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'
Input drops: 0 [0]
Input framing errors: 0 [0]
Policed discards: 0 [0]
L3 incompletes: 0 [0]
L2 channel errors: 0 [0]
L2 mismatch timeouts: 0 [0]
Carrier transitions: 0 [0]
Output errors: 0 [0]
Output drops: 0 [0]
Aged packets: 0 [0]
Active alarms: LINK
Active defects: LINK
Input MAC/Filter statistics:
Unicast packets 0 [0]
Broadcast packets 0 Multicast packet [0]
Interface warnings:
o Outstanding LINK alarm

monitor interface traffic

user@host> monitor interface traffic

host name Seconds: 15 Time: 12:31:09

<table>
<thead>
<tr>
<th>Interface</th>
<th>Link</th>
<th>Input packets (pps)</th>
<th>Output packets (pps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>so-1/0/0</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>so-1/1/0</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>so-1/1/1</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>so-1/1/2</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>so-1/1/3</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>t3-1/2/0</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>t3-1/2/1</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>t3-1/2/2</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>t3-1/2/3</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>so-2/0/0</td>
<td>Up</td>
<td>211035 (1)</td>
<td>36778 (0)</td>
</tr>
<tr>
<td>so-2/0/1</td>
<td>Up</td>
<td>192753 (1)</td>
<td>36782 (0)</td>
</tr>
<tr>
<td>so-2/0/2</td>
<td>Up</td>
<td>211020 (1)</td>
<td>36779 (0)</td>
</tr>
<tr>
<td>so-2/0/3</td>
<td>Up</td>
<td>211029 (1)</td>
<td>36776 (0)</td>
</tr>
<tr>
<td>so-2/1/0</td>
<td>Up</td>
<td>189378 (1)</td>
<td>36349 (0)</td>
</tr>
<tr>
<td>so-2/1/1</td>
<td>Down</td>
<td>0 (0)</td>
<td>18747 (0)</td>
</tr>
<tr>
<td>so-2/1/2</td>
<td>Down</td>
<td>0 (0)</td>
<td>16078 (0)</td>
</tr>
<tr>
<td>so-2/1/3</td>
<td>Up</td>
<td>0 (0)</td>
<td>80338 (0)</td>
</tr>
<tr>
<td>at-2/3/0</td>
<td>Up</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>at-2/3/1</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
</tbody>
</table>

Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D

monitor interface traffic (QFX3500 Switch)

user@switch> monitor interface traffic

switch Seconds: 7 Time: 16:04:37

<table>
<thead>
<tr>
<th>Interface</th>
<th>Link</th>
<th>Input packets (pps)</th>
<th>Output packets (pps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ge-0/0/0</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>ge-0/0/1</td>
<td>Up</td>
<td>392187 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>ge-0/0/2</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>ge-0/0/3</td>
<td>Down</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
</tbody>
</table>
### monitor interface traffic detail (QFX3500 Switch)

```
user@switch> monitor interface traffic detail

switch

Time: 16:03:02

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
<th>Link</th>
<th>Input packets</th>
<th>(pps)</th>
<th>Output packets</th>
<th>(pps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ge-0/0/0</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/1</td>
<td>Up</td>
<td>392183</td>
<td>(0)</td>
<td></td>
<td>392166</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/2</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/3</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/4</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/5</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/6</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/7</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/8</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/9</td>
<td>Up</td>
<td>392181</td>
<td>(0)</td>
<td></td>
<td>392168</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/10</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/11</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/12</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/13</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/14</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/15</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/16</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/17</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/18</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/19</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/20</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/21</td>
<td>Down</td>
<td></td>
<td>0</td>
<td>(0)</td>
<td>0</td>
<td>(0)</td>
</tr>
<tr>
<td>ge-0/0/22</td>
<td>Up</td>
<td>392169</td>
<td>(0)</td>
<td></td>
<td>392184</td>
<td>(1)</td>
</tr>
<tr>
<td>ge-0/0/23</td>
<td>Up</td>
<td>392182</td>
<td>(0)</td>
<td></td>
<td>392170</td>
<td>(0)</td>
</tr>
<tr>
<td>vcp-0</td>
<td>Down</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>vcp-1</td>
<td>Down</td>
<td></td>
<td>0</td>
<td></td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
```


<table>
<thead>
<tr>
<th>Interface</th>
<th>Status</th>
<th>IPv6</th>
<th>64-bit</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>ae0</td>
<td>Down</td>
<td>0</td>
<td>(0)</td>
<td>0</td>
</tr>
<tr>
<td>bme0</td>
<td>Up</td>
<td>0</td>
<td>(0)</td>
<td>156893</td>
</tr>
</tbody>
</table>
monitor start

Syntax

monitor start filename

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.

Description

Start displaying the system log or trace file and additional entries being added to those files.

Options

filename—Specific log or trace file.

Additional Information

Log files are generated by the routing protocol process or by system logging. The log files generated by system logging are configured with the syslog statement at the [edit system] hierarchy level and the options statement at the [edit routing-options] hierarchy level. The trace files generated by the routing protocol process are configured with traceoptions statements at the [edit routing-options], [edit interfaces], and [edit protocols protocol] hierarchy levels.

NOTE: To monitor a log file within a logical system, issue the monitor start logical-system-name/filename command.

Required Privilege

trace

Related Documentation

• monitor list
• monitor stop on page 121

List of Sample Output

monitor start on page 120

Output Fields

Table 7 on page 119 describes the output fields for the monitor start command. Output fields are listed in the approximate order in which they appear.

Table 7: monitor start Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em><strong>filename</strong></em></td>
<td>Name of the file from which entries are being displayed. This line is displayed initially and when the command switches between log files.</td>
</tr>
<tr>
<td>Date and time</td>
<td>Timestamp for the log entry.</td>
</tr>
</tbody>
</table>
Sample Output

monitor start

```
user@host> monitor start system-log

*** system-log***
Jul 20 15:07:34 hang sshd[5845]: log: Generating 768 bit RSA key.
Jul 20 15:07:35 hang sshd[5845]: log: Connection from 204.69.248.180 port 912
Jul 20 15:07:37 hang sshd[5845]: log: ROOT LOGIN as 'root' from host.example.com
Jul 20 15:07:37 hang sshd[5845]: log: Closing connection to 204.69.248.180
```
### monitor stop

<table>
<thead>
<tr>
<th>Syntax</th>
<th>monitor stop <code>filename</code></th>
</tr>
</thead>
</table>

**Release Information**  
Command introduced before Junos OS Release 7.4.  
Command introduced in Junos OS Release 9.0 for EX Series switches.

**Description**  
Stop displaying the system log or trace file.

**Options**  
`filename`—Specific log or trace file.

**Additional Information**  
Log files are generated by the routing protocol process or by system logging. The log files generated by system logging are those configured with the `syslog` statement at the `[edit system]` hierarchy level and the `options` statement at the `[edit routing-options]` hierarchy level. The trace files generated by the routing protocol process are those configured with `traceoptions` statements at the `[edit routing-options]`, `[edit interfaces]`, and `[edit protocols protocol]` hierarchy levels.

**Required Privilege Level**  
trace

**Related Documentation**  
- `monitor list`
- `monitor start on page 119`

**List of Sample Output**  
`monitor stop on page 121`

**Output Fields**  
This command produces no output.

**Sample Output**

```
monitor stop

user@host> monitor stop
```
ping

List of Syntax

Syntax on page 122
Syntax (QFX Series) on page 122
Syntax (Junos OS Evolved) on page 123

Syntax

```xml
ping host
  <bypass-routing>
  <ce-ip destination-ip-address instance routing-instance-name source-ip source-ip-address>
  <count requests>
  <do-not-fragment>
  <inet | inet6>
  <interface source-interface>
  <interval seconds>
  <no-resolve>
  <pattern string>
  <rapid>
  <record-route>
  <routing-instance routing-instance-name>
  <size bytes>
  <source source-address>
  <tos type-of-service>
  <ttl value>
  <verbose>
  <wait seconds>
```

Syntax (QFX Series)

```xml
ping host
  <bypass-routing>
  <count requests>
  <detail>
  <do-not-fragment>
  <inet>
  <interface source-interface>
  <interval seconds>
  <logical-system logical-system-name>
  <loose-source value>
  <mac-address mac-address>
  <no-resolve>
  <pattern string>
  <rapid>
  <record-route>
  <routing-instance routing-instance-name>
  <size bytes>
  <source source-address>
  <strict>
  <strict-source value>
  <tos type-of-service>
  <ttl value>
  <verbose>
  <wait seconds>
```
Syntax (Junos OS Evolved)

```plaintext
ping host
<bypass-routing>
<ce-ip destination-ip-address instance routing-instance-name source-ip source-ip-address>
<count requests>
<do-not-fragment>
<inet | inet6>
<interface source-interface>
<interval seconds>
<no-resolve>
<pattern string>
<rapid>
<record-route>
<routing-instance routing-instance-name>
<size bytes>
<source source-address>
<tos type-of-service>
<ttl value>
<verbose>
<wait seconds>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
ce-ip option introduced in Junos OS Release 17.3 for MX Series routers with MPC and MIC interfaces.
The following options are deprecated for Junos OS Evolved Release 18.3R1: detail, logical-system, loose-source, mac-address, strict, strict-source, and vpls.

Description

Check host reachability and network connectivity. The `ping` command sends Internet Control Message Protocol (ICMP) ECHO_REQUEST messages to elicit ICMP ECHO_RESPONSE messages from the specified host. Press Ctrl+c to interrupt a ping command.

Options

- **host**—IP address or hostname of the remote system to ping.

  - **bypass-routing**—(Optional) Bypass the normal routing tables and send ping requests directly to a system on an attached network. If the system is not on a directly attached network, an error is returned. Use this option to ping a local system through an interface that has no route through it.

- **ce-ip destination-ip-address instance routing-instance-name source-ip source-ip-address**—(MX Series routers with MPC and MIC interfaces only) (Optional) Check the connectivity information of customer edge (CE) devices, such as reachability, attachment points, and MAC addresses, from a provider edge (PE) device in a virtual private LAN service (VPLS), hierarchical VPLS (H-VPLS), and Ethernet VPN (EVPN) network. The ce-ip option is based on the LSP ping infrastructure, where the ping utility is extended to use the CE device IP address as the target host and the PE device loopback address as the source for a specific VPLS or EVPN routing instance.
destination-ip-address—IPv4 address of the CE device to ping.

instance routing-instance-name—Name of the VPLS or EVVPN routing instance. The command output displays the connectivity information of the CE device based on the configured routing instance type.

source-ip source-ip-address—Loopback address of the PE device.

count requests—(Optional) Number of ping requests to send. The range of values is 1 through 2,000,000,000. The default value is an unlimited number of requests.

detail—(Optional) This option is not supported for Junos OS Evolved Release 18.3R1. Include in the output the interface on which the ping reply was received.

do-not-fragment—(Optional) Set the do-not-fragment (DF) flag in the IP header of the ping packets.

For Junos OS Evolved Release 18.3R1, IPv6 ping does not have do-not-fragment support. The ping command is identified as IPv6 Ping when destination is IPv6 address or inet6 option is used.

For Junos OS IPv6 packets, this option disables fragmentation.

inet—(Optional) Ping Packet Forwarding Engine IPv4 routes.

inet6—(Optional) Ping Packet Forwarding Engine IPv6 routes.

interface source-interface—(Optional) Interface to use to send the ping requests.

interval seconds—(Optional) How often to send ping requests. The range of values, in seconds, is 1 through infinity. The default value is 1.

logical-system logical-system-name—(Optional) Name of logical system from which to send the ping requests.

Alternatively, enter the set cli logical-system logical-system-name command and then run the ping command. To return to the main router or switch, enter the clear cli logical-system command.

loose-source value—(Optional) Intermediate loose source route entry (IPv4). Open a set of values.

---

NOTE: In Junos OS Release 11.1 and later, when issuing the ping command for an IPv6 route with the do-not-fragment option, the maximum ping packet size is calculated by subtracting 48 bytes (40 bytes for the IPv6 header and 8 bytes for the ICMP header) from the MTU. Therefore, if the ping packet size (including the 48-byte header) is greater than the MTU, the ping operation might fail.
`mac-address mac-address`—(Optional) Ping the physical or hardware address of the remote system you are trying to reach.

`noResolve`—(Optional) Do not attempt to determine the hostname that corresponds to the IP address.

`pattern string`—(Optional) Specify a hexadecimal fill pattern to include in the ping packet.

`rapid`—(Optional) Send ping requests rapidly. The results are reported in a single message, not in individual messages for each ping request. By default, five ping requests are sent before the results are reported. To change the number of requests, include the `count` option.

`record-route`—(Optional) Record and report the packet’s path (IPv4).

`routing-instance routing-instance-name`—(Optional) Name of the routing instance for the ping attempt. For Junos OS Evolved, the `routing-instance` option supports only `mgmt_junos`.

`size bytes`—(Optional) Size of ping request packets. The range of values, in bytes, is 0 through 65,468. The default value is 56, which is effectively 64 bytes because 8 bytes of ICMP header data are added to the packet.

`source source-address`—(Optional) IP address of the outgoing interface. This address is sent in the IP source address field of the ping request. If this option is not specified, the default address is usually the loopback interface (`lo.0`).

`strict`—(Optional) Use the strict source route option (IPv4).

`strict-source value`—(Optional) Intermediate strict source route entry (IPv4). Open a set of values.

`tos type-of-service`—(Optional) Set the type-of-service (ToS) field in the IP header of the ping packets. The range of values is 0 through 255.

If the device configuration includes the `dscp-code-point value` statement at the [edit class-of-service host-outbound-traffic] hierarchy level, the configured DSCP value overrides the value specified in this command option. In this case, the ToS field of ICMP echo request packets sent on behalf of this command carries the DSCP value specified in the `dscp-code-point` configuration statement instead of the value you specify in this command option.

`ttl value`—(Optional) Time-to-live (TTL) value to include in the ping request (IPv6). The range of values is 0 through 255.

`verbose`—(Optional) Display detailed output.

`vpls instance-name`—(Optional) Ping the instance to which this VPLS belongs.

`wait seconds`—(Optional) Maximum wait time, in seconds, after the final packet is sent. If this option is not specified, the default delay is 10 seconds. If this option is used without the `count` option, a default count of 5 packets is used.
Required Privilege Level

- network

Related Documentation

- Rate Limiting ICMPv4 and ICMPv6 Traffic
- Pinging Customer Edge Device IP Address

List of Sample Output

- ping ce-ip <destination-ip-address> instance <routing-instance-name> source-ip <source-ip-address> (EVPN) on page 126
- ping ce-ip <destination-ip-address> instance <routing-instance-name> source-ip <source-ip-address> (VPLS) on page 126
- ping hostname on page 126
- ping hostname rapid on page 127
- ping hostname size count on page 127

Output Fields

When you enter this command, you are provided feedback on the status of your request. An exclamation point (!) indicates that an echo reply was received. A period (.) indicates that an echo reply was not received within the timeout period. An x indicates that an echo reply was received with an error code. These packets are not counted in the received packets count. They are accounted for separately.

When pinging a nonexistent route, the display output of ping command does not print the number of packets sent or received or the packet loss.

Sample Output

**ping ce-ip <destination-ip-address> instance <routing-instance-name> source-ip <source-ip-address> (EVPN)**

```
user@host> ping ce-ip 10.0.0.4 instance foo source-ip 127.0.0.1
--- ce-ip ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

**ping ce-ip <destination-ip-address> instance <routing-instance-name> source-ip <source-ip-address> (VPLS)**

```
user@host> ping ce-ip 10.0.0.4 instance foo source-ip 127.0.0.1
! -> PE2|foo|vpls|ge-0/0/2.100|12:23:ab:98:34:02
! -> PE2|foo|vpls|ge-0/0/2.100|12:23:ab:98:34:02
! -> PE2|foo|vpls|ge-0/0/2.100|12:23:ab:98:34:02
! -> PE2|foo|vpls|ge-0/0/2.100|12:23:ab:98:34:02
! -> PE2|foo|vpls|ge-0/0/2.100|12:23:ab:98:34:02
--- ce-ip ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
```

**ping hostname**

```
user@host> ping device1.example.com
```
PING device1.example.com (192.0.2.0): 56 data bytes
64 bytes from 192.0.2.0: icmp_seq=0 ttl=253 time=1.028 ms
64 bytes from 192.0.2.0: icmp_seq=1 ttl=253 time=1.053 ms
64 bytes from 192.0.2.0: icmp_seq=2 ttl=253 time=1.025 ms
64 bytes from 192.0.2.0: icmp_seq=3 ttl=253 time=1.098 ms
64 bytes from 192.0.2.0: icmp_seq=4 ttl=253 time=1.032 ms
64 bytes from 192.0.2.0: icmp_seq=5 ttl=253 time=1.044 ms
^C [abort]

ping hostname rapid

user@host> ping device1.example.com rapid
PING device1.example.com (192.0.2.0): 56 data bytes
!!!!!
--- device1.example.com ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.956/0.974/1.025/0.026 ms

ping hostname size count

user@host> ping device1.example.com size 200 count 5
PING device1.example.com (192.0.2.0): 200 data bytes
208 bytes from 192.0.2.0: icmp_seq=0 ttl=253 time=1.759 ms
208 bytes from 192.0.2.0: icmp_seq=1 ttl=253 time=2.075 ms
208 bytes from 192.0.2.0: icmp_seq=2 ttl=253 time=1.843 ms
208 bytes from 192.0.2.0: icmp_seq=3 ttl=253 time=1.803 ms
208 bytes from 192.0.2.0: icmp_seq=4 ttl=253 time=17.898 ms
--- device1.example.com ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.759/5.075/17.898 ms

Output for Junos OS Evolved:

user@host> ping 40.0.0.2 count 20 size 500
connect: No route to host

Output for Junos OS:

user@host> ping 40.0.0.2 count 20 size 500
Aug 02 12:56:56 [INFO ] Step 2: Host and Transit ping has to fail
Aug 02 12:56:56 [TRACE] [R0 evo-pxb-b] [cmd] run ping 40.0.0.2 rapid count 50 size 500
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] PING 40.0.0.2 (40.0.0.2): 500 data bytes
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] .ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
Aug 02 12:57:21 [TRACE] [R0 evo-pxb-b] ping: sendto: No route to host
**show ipv6 neighbors**

**Syntax**

```plaintext
show ipv6 neighbors
<flags>
<hostname host-name>
<interface interface-name>
<logical-system logical-system-name>
<reference-count count>
<tenant name>
<vpn vpn-name>
```

**Release Information**

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.3 for EX Series switches.
Command introduced in Junos OS Release 12.2 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
`flags`, `hostname`, `interface`, `logical-system`, `reference-count`, `tenant`, and `vpn` options added in Junos OS Release 18.3.

**Description**

Display information about the IPv6 neighbor cache.

---

**NOTE:** Starting with Junos OS Release 16.1, `show ipv6 neighbors` command does not display the underlying ifl information if `enhanced-convergence` statement at `[edit irb unit unit-number]` hierarchy level and `enhanced-ip` statement at `[edit chassis network-services]` hierarchy level is configured for the destination interface IRB.

**Options**

- **none**—Display the entries in the IPv6 table.
- **flags**—(Optional) Display the flags set, if any.
- **hostname host-name**—(Optional) Display the hostname.
- **interface interface-name**—(Optional) Display information about IPv6 for the specified logical interface
- **logical-system logical-system-name**—(Optional) Display the IPv6 entries for the specified logical system; only available on the main router context.
- **reference-count count**—(Optional) Display the IPv6 next-hop reference count.
- **tenant name**—(Optional) Display the name of the tenant.
- **vpn vpn-name**—(Optional) Display entries in the IPv6 table for the specified virtual private network’s (VPN) routing table.
List of Sample Output

- clear ipv6 neighbors on page 104
- show ipv6 neighbors on page 129
- show ipv6 neighbors on page 129

Output Fields

Table 8 on page 129 describes the output fields for the `show ipv6 neighbors` command. Output fields are listed in the approximate order in which they appear.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv6 Address</td>
<td>Name of the IPv6 interface.</td>
</tr>
<tr>
<td>Linklayer Address</td>
<td>Link-layer address.</td>
</tr>
<tr>
<td>State</td>
<td>State of the link: up, down, incomplete, reachable, stale, or unreachable.</td>
</tr>
<tr>
<td>Exp</td>
<td>Number of seconds until the entry expires.</td>
</tr>
<tr>
<td>Rtr</td>
<td>Whether the neighbor is a routing device: yes or no.</td>
</tr>
<tr>
<td>Secure</td>
<td>Whether this entry was created using the Secure Neighbor Discovery (SEND) protocol: yes or no.</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the interface.</td>
</tr>
</tbody>
</table>

Sample Output

`show ipv6 neighbors`

```
user@host> show ipv6 neighbors
IPv6 Address     Linklayer Address  State       Exp  Rtr  Secure
----------------------------------------------------------------
2001:db8:0:1:2a0:a514:0:24c 00:05:85:8f:c8:bd  stale   546 yes no
fe-1/2/0.1          fe80::2a0:a514:0:24c 00:05:85:8f:c8:bd  stale   258 yes no
fe-1/2/0.1          fe80::2a0:a514:0:64c 00:05:85:8f:c8:bd  stale   111 yes no
fe-1/2/1.5          fe80::2a0:a514:0:a4c 00:05:85:8f:c8:bd  stale   327 yes no
```

- show ipv6 neighbors

The command displaying the underlying l2 ifl information when enhanced-convergence statement and enhanced-ip statement is not configured.
<table>
<thead>
<tr>
<th>IPv6 Address</th>
<th>Linklayer Address</th>
<th>State</th>
<th>Exp Rtr</th>
<th>Secure</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>23::23:0:0:2</td>
<td>00:00:23:00:00:02</td>
<td>reachable</td>
<td>0</td>
<td>no</td>
<td>irb.0</td>
</tr>
</tbody>
</table>

The command not displaying the underlying l2if information when `enhanced-convergence` statement and `enhanced-ip` statement is configured.
**show ipv6 router-advertisement**

**Syntax**

```
show ipv6 router-advertisement
<conflicts>
<interface interface>
<logical-system (all | logical-system-name)>
<prefix prefix/prefix length>
```

**Release Information**

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 12.2 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

**Description**

Display information about IPv6 router advertisements, including statistics about messages sent and received on interfaces, and information received from advertisements from other routers.

**Options**

- `none`—Display all IPv6 router advertisement information for all interfaces.
- `conflicts`—(Optional) Display only the IPv6 router advertisement information that is conflicting.
- `interface interface`—(Optional) Display IPv6 router advertisement information for the specified interface.
- `logical-system (all | logical-system-name)`—(Optional) Perform this operation on all logical systems or on a particular logical system.
- `prefix prefix/prefix length`—(Optional) Display IPv6 router advertisement information for the specified prefix.

**Additional Information**

The display identifies conflicting information by enclosing the value the router is advertising in brackets.

**Required Privilege Level**

view

**Related Documentation**

- `clear ipv6 router-advertisement` on page 105

**List of Sample Output**

- `show ipv6 router-advertisement` on page 132
- `show ipv6 router-advertisement conflicts` on page 133
- `show ipv6 router-advertisement prefix` on page 133

**Output Fields**

Table 9 on page 132 describes the output fields for the `show ipv6 router-advertisement` command. Output fields are listed in the approximate order in which they appear.
### Table 9: show ipv6 router-advertisement Output Fields

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Name of the interface.</td>
</tr>
<tr>
<td>Advertisements sent</td>
<td>Number of router advertisements sent and the elapsed time since they were sent.</td>
</tr>
<tr>
<td>Solicits received</td>
<td>Number of solicitation messages received.</td>
</tr>
<tr>
<td>Advertisements received</td>
<td>Number of router advertisements received.</td>
</tr>
<tr>
<td>Advertisements from</td>
<td>Names of interfaces from which router advertisements have been received and the elapsed time since the last one was received.</td>
</tr>
<tr>
<td>Managed</td>
<td>Managed address configuration flag: 0 (stateless) or 1 (stateful).</td>
</tr>
<tr>
<td>Other configuration</td>
<td>Other stateful configuration flag: 0 (stateless) or 1 (stateful).</td>
</tr>
<tr>
<td>Reachable time</td>
<td>Time that a node identifies a neighbor as reachable after receiving a reachability confirmation, in milliseconds.</td>
</tr>
<tr>
<td>Default lifetime</td>
<td>Default lifetime, in seconds: from 0 seconds to 18.2 hours. A setting of 0 indicates that the router is not a default router.</td>
</tr>
<tr>
<td>Retransmit timer</td>
<td>Time between retransmitted Neighbor Solicitation messages, in milliseconds.</td>
</tr>
<tr>
<td>Current hop limit</td>
<td>Configured current hop limit.</td>
</tr>
<tr>
<td>Prefix</td>
<td>Name and length of the prefix.</td>
</tr>
<tr>
<td>Valid lifetime</td>
<td>How long the prefix remains valid for onlink determination.</td>
</tr>
<tr>
<td>Preferred lifetime</td>
<td>How long the prefix generated by stateless autoconfiguration remains preferred.</td>
</tr>
<tr>
<td>On link</td>
<td>Onlink flag: 0 (not onlink) or 1 (onlink).</td>
</tr>
<tr>
<td>Autonomous</td>
<td>Autonomous address configuration flag: 0 (not autonomous) or 1 (autonomous).</td>
</tr>
</tbody>
</table>

### Sample Output

```plaintext
show ipv6 router-advertisement

user@host> show ipv6 router-advertisement
Interface: fe-0/1/1.0
  Advertisements sent: 0
  Solicits received: 0
Interface: fxp0.0
  Advertisements sent: 0
  Solicits received: 0
```
Advertisements received: 1
Advertisement from fe80::2d0:b7ff:fe1e:7b0e, heard 00:00:13 ago
Managed: 0
Other configuration: 0 [1]
  Reachable time: 0 ms
  Default lifetime: 1800 sec
  Retransmit timer: 0 ms
  Current hop limit: 64

show ipv6 router-advertisement conflicts

  user@host> show ipv6 router-advertisement conflicts
  
  Interface: fxp0.0
  Advertisement from fe80::2d0:b7ff:fe1e:7b0e, heard 00:01:08 ago
  Other configuration: 0 [1]

show ipv6 router-advertisement prefix

  user@host> show ipv6 router-advertisement prefix 2001:db8:8040::/16
  
  Interface: fe-0/1/3.0
  Advertisements sent: 3, last sent 00:04:11 ago
  Solicits received: 0
  Advertisements received: 3
  Advertisement from fe80::290:69ff:fe9a:5403, heard 00:00:05 ago
  Managed: 0
  Other configuration: 0
  Reachable time: 0 ms
  Default lifetime: 180 sec [1800 sec]
  Retransmit timer: 0 ms
  Current hop limit: 64
  Prefix: 2001:db8:8040::/64
    Valid lifetime: 2592000 sec
    Preferred lifetime: 604800 sec
    On link: 1
    Autonomous: 1
show log

List of Syntax

Syntax (QFX Series and OCX Series) on page 134
Syntax (TX Matrix Router) on page 134

Syntax

show log

<filename | user <username>>

Syntax (QFX Series and OCX Series)

show log filename
<device-type (device-id | device-alias)>

Syntax (TX Matrix Router)

show log
<all-lcc | lcc number | scc>
<filename | user <username>>

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Option device-type (device-id | device-alias) is introduced in Junos OS Release 13.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

List log files, display log file contents, or display information about users who have logged in to the router or switch.

NOTE: On MX Series routers, modifying a configuration to replace a service interface with another service interface is treated as a catastrophic event. When you modify a configuration, the entire configuration associated with the service interface—including NAT pools, rules, and service sets—is deleted and then re-created for the newly specified service interface. If there are active sessions associated with the service interface that is being replaced, these sessions are deleted and the NAT pools are then released, which leads to the generation of the NAT_POOL_RELEASE system log messages. However, because NAT pools are already deleted as a result of the catastrophic configuration change and no longer exist, the NAT_POOL_RELEASE system log messages are not generated for the changed configuration.

Options

none—List all log files.

<all-lcc | lcc number | scc>—(Routing matrix only) (Optional) Display logging information about all T640 routers (or line-card chassis) or a specific T640 router (replace
number with a value from 0 through 3) connected to a TX Matrix router. Or, display logging information about the TX Matrix router (or switch-card chassis).

device-type—(QFabric system only) (Optional) Display log messages for only one of the following device types:

- director-device—Display logs for Director devices.
- interconnect-device—Display logs for Interconnect devices.
- node-device—Display logs for Node devices.

NOTE: If you specify the device-type optional parameter, you must also specify either the device-id or device-alias optional parameter.

(device-id | device-alias)—If a device type is specified, display logs for a device of that type. Specify either the device ID or the device alias (if configured).

filename—(Optional) Display the log messages in the specified log file. For the routing matrix, the filename must include the chassis information.

NOTE: The filename parameter is mandatory for the QFabric system. If you did not configure a syslog filename, specify the default filename of messages.

user <username>—(Optional) Display logging information about users who have recently logged in to the router or switch. If you include username, display logging information about the specified user.

Required Privilege Level trace

Related Documentation • syslog (System)

List of Sample Output show log on page 136
show log filename on page 136
show log filename (QFabric System) on page 136
show log user on page 137
Sample Output

```bash
show log

user@host> show log

<table>
<thead>
<tr>
<th>Mode</th>
<th>User</th>
<th>Group</th>
<th>Permissions</th>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Time</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>211663</td>
<td>Oct 1 19:44</td>
<td>dcd</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>999947</td>
<td>Oct 1 19:41</td>
<td>dcd.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>999994</td>
<td>Oct 1 17:48</td>
<td>dcd.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>238815</td>
<td>Oct 1 19:44</td>
<td>rpd</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1049098</td>
<td>Oct 1 18:00</td>
<td>rpd.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1061095</td>
<td>Oct 1 12:13</td>
<td>rpd.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1052026</td>
<td>Oct 1 06:08</td>
<td>rpd.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1056309</td>
<td>Sep 30 18:21</td>
<td>rpd.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1056371</td>
<td>Sep 30 14:36</td>
<td>rpd.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1056301</td>
<td>Sep 30 10:30</td>
<td>rpd.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1056350</td>
<td>Sep 30 07:04</td>
<td>rpd.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>1048876</td>
<td>Sep 30 03:21</td>
<td>rpd.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-rw-r--r--</td>
<td>root</td>
<td>bin</td>
<td>19656</td>
<td>Oct 1 19:37</td>
<td>wtmp</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```bash
show log filename

user@host> show log rpd

Oct 1 18:00:18 trace_on: Tracing to ?/var/log/rpd? started
Oct 1 18:00:18 EVENT <MTU> ds-5/2/0.0 index 24 <Broadcast PointToPoint Multicast
Oct 1 18:00:18
Oct 1 18:00:19 KRT recv len 56 V9 seq 148 op add Type route/if af 2 addr
192.0.2.21 nhop type local nhop 192.0.2.21
Oct 1 18:00:19 KRT recv len 56 V9 seq 149 op add Type route/if af 2 addr
192.0.2.22 nhop type unicast nhop 192.0.2.22
Oct 1 18:00:19 KRT recv len 48 V9 seq 150 op add Type ifaddr index 24 devindex 43
Oct 1 18:00:19 KRT recv len 144 V9 seq 151 op chnge Type ifdev devindex 44
Oct 1 18:00:19 KRT recv len 144 V9 seq 152 op chnge Type ifdev devindex 45
Oct 1 18:00:19 KRT recv len 144 V9 seq 153 op chnge Type ifdev devindex 46
Oct 1 18:00:19 KRT recv len 1272 V9 seq 154 op chnge Type ifdev devindex 47 ...
```

show log filename (QFabric System)

```bash
user@qfabric> show log messages

Mar 28 18:00:06 qfabric chassisd: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:06 ED1486
chassisd: CHASSISD_SNMP_TRAP10: SNMP trap generated: FRU power on
(jnxFruContentsIndex 8, jnxFruL1Index 1, jnxFruL2Index 1, jnxFruL3Index 0,
jnxFruName PIC: 48x 10G-SFP+ @ 0/0/*, jnxFruType 11, jnxFruSlot 0,
jnxFruOfflineReason 2, jnxFruLastPowerOff 0, jnxFruLastPowerOn 2159)
Mar 28 18:00:07 qfabric chassisd: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:07 ED1486
chassisd: CHASSISD_SNMP_TRAP10: SNMP trap generated: FRU power on
(jnxFruContentsIndex 8, jnxFruL1Index 1, jnxFruL2Index 1, jnxFruL3Index 0,
jnxFruName PIC: @ 0/1/*, jnxFruType 11, jnxFruSlot 0, jnxFruOfflineReason 2,
jnxFruLastPowerOff 0, jnxFruLastPowerOn 2191)
Mar 28 18:00:07 qfabric chassisd: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:07 ED1492
chassisd: CHASSISD_SNMP_TRAP10: SNMP trap generated: FRU power on
(jnxFruContentsIndex 8, jnxFruL1Index 1, jnxFruL2Index 1, jnxFruL3Index 0,
jnxFruName PIC: 48x 10G-SFP+ @ 0/0/*, jnxFruType 11, jnxFruSlot 0,
jnxFruOfflineReason 2, jnxFruLastPowerOff 0, jnxFruLastPowerOn 242726)
```
show log user

user@host> show log user

<table>
<thead>
<tr>
<th>User</th>
<th>Terminal</th>
<th>Host</th>
<th>Date/Time</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>usera</td>
<td>mg2546</td>
<td>Thu Oct 1 19:37</td>
<td>Thu Oct 1 19:37 - 19:38 (00:01)</td>
<td>still logged in</td>
</tr>
<tr>
<td>usera</td>
<td>mg2529</td>
<td>Thu Oct 1 19:08</td>
<td>Thu Oct 1 19:08 - 19:36 (00:28)</td>
<td></td>
</tr>
<tr>
<td>usera</td>
<td>mg2518</td>
<td>Thu Oct 1 18:53</td>
<td>Thu Oct 1 18:53 - 18:55 (00:02)</td>
<td></td>
</tr>
<tr>
<td>root</td>
<td>ttyp1</td>
<td>Wed Sep 30 01:03</td>
<td>Wed Sep 30 01:03 - 01:22 (00:19)</td>
<td></td>
</tr>
<tr>
<td>root</td>
<td>192.0.2.0</td>
<td>Wed Sep 30 01:03</td>
<td>Wed Sep 30 01:03 - 01:22 (00:19)</td>
<td></td>
</tr>
</tbody>
</table>
# traceroute

<table>
<thead>
<tr>
<th>List of Syntax</th>
<th>Syntax on page 138</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>Syntax (QFX Series and OCX Series) on page 138</td>
</tr>
</tbody>
</table>

**Syntax**

```
traceroute host
<as-number-lookup>
<bypass-routing>
<clns>
<gateway address>
/inet | inet6>
.interface interface-name>
<monitor host>
<mpls (ldp FEC address | rsvp label-switched-path-name)>
<no-resolve>
<routing-instance routing-instance-name>
<source source-address>
<tos value>
<ttl value>
<wait seconds>
```

**Syntax (QFX Series and OCX Series)**

```
traceroute host
<as-number-lookup>
<bypass-routing>
<gateway address>
/inet>
/inet6>
.interface interface-name>
<monitor host>
<no-resolve>
<routing-instance routing-instance-name>
<source source-address>
<tos value>
<ttl value>
<wait seconds>
```

**Release Information**


**Description**

Display the route that packets take to a specified network host. Use **traceroute** as a debugging tool to locate points of failure in a network.
**Options**

- **host**—IP address or name of remote host.
- **as-number-lookup**—(Optional) Display the autonomous system (AS) number of each intermediate hop on the path from the host to the destination.
- **bypass-routing**—(Optional) Bypass the normal routing tables and send requests directly to a system on an attached network. If the system is not on a directly attached network, an error is returned. Use this option to display a route to a local system through an interface that has no route through it.
- **clns**—(Optional) Trace the route belonging to Connectionless Network Service (CLNS).
- **gateway address**—(Optional) Address of a router or switch through which the route transits.
- **inet | inet6**—(Optional) Trace the route belonging to IPv4 or IPv6, respectively.
- **interface interface-name**—(Optional) Name of the interface over which to send packets.
- **logical-system (all | logical-system-name)**—(Optional) This option is not supported for Junos OS Evolved Release 18.3R1. Perform this operation on all logical systems or on a particular logical system.
- **monitor host**—(Optional) Display real-time monitoring information for the specified host.
- **monitorhost**—(Optional) Perform this operation to display real-time monitoring information.
- **monitorhost**—(Optional) Perform this operation to display real-time monitoring information.
- **monitorhost**—(Optional) Perform this operation to display real-time monitoring information.
- **mpls (ldp FEC address | rsvp label-switched-path name)**—(Optional) See traceroute mpls ldp and traceroute mpls rsvp.
- **next-hop**—The next-hop through which to send packets to a destination.
- **no-resolve**—(Optional) Do not attempt to determine the hostname that corresponds to the IP address.
- **propagate-ttl**—(Optional) On the PE routing device, use this option to view locally generated Routing Engine transit traffic. This is applicable for MPLS L3VPN traffic only.

Use for troubleshooting, when you want to view hop-by-hop information from the local provider router to the remote provider router, when TTL decrementing is disabled on the core network using the **no-proagate-ttl** configuration statement.

**NOTE:** Using propagate-ttl with traceroute on the CE router does not show hop-by-hop information.
**routing-instance routing-instance-name**—(Optional) Name of the routing instance for the traceroute attempt.

**source source-address**—(Optional) Source address of the outgoing traceroute packets.

**tos value**—(Optional) Value to include in the IP type-of-service (ToS) field. The range of values is 0 through 255.

**ttl value**—(Optional) Maximum time-to-live value to include in the traceroute request. The range of values is 0 through 128.

**wait seconds**—(Optional) Maximum time to wait for a response to the traceroute request.

**Required Privilege Level**

network

**Related Documentation**

• traceroute monitor

**List of Sample Output**

traceroute on page 141
traceroute as-number-lookup host on page 141
traceroute no-resolve on page 141
traceroute propogate-ttl on page 141
traceroute (Between CE Routers, Layer 3 VPN) on page 142
traceroute (Through an MPLS LSP) on page 142

**Output Fields**

Table 10 on page 140 describes the output fields for the traceroute command. Output fields are listed in the approximate order in which they appear.

*Table 10: traceroute Output Fields*

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>traceroute to</td>
<td>IP address of the receiver.</td>
</tr>
<tr>
<td>hops max</td>
<td>Maximum number of hops allowed.</td>
</tr>
<tr>
<td>byte packets</td>
<td>Size of packets being sent.</td>
</tr>
<tr>
<td>number-of-hops</td>
<td>Number of hops from the source to the named router or switch.</td>
</tr>
<tr>
<td>router-name</td>
<td>Name of the router or switch for this hop.</td>
</tr>
<tr>
<td>address</td>
<td>Address of the router or switch for this hop.</td>
</tr>
<tr>
<td>Round trip time</td>
<td>Average round-trip time, in milliseconds (ms).</td>
</tr>
</tbody>
</table>
Sample Output

traceroute

```
user@host> traceroute santacruz
traceroute to host1.example.com (10.156.169.254), 30 hops max, 40 byte packets
 1  blue23 (10.168.1.254)  2.370 ms  2.853 ms  0.367 ms
 2  red14  (10.168.255.250)  0.778 ms  2.937 ms  0.446 ms
 3  yellow (10.156.169.254)  7.737 ms  89.905 ms  0.834 ms
```

traceroute as-number-lookup host

```
user@host> traceroute as-number-lookup 10.100.1.1
traceroute to 10.100.1.1 (10.100.1.1), 30 hops max, 40 byte packets
 1  10.39.1.1 (10.39.1.1)  0.779 ms  0.728 ms  0.562 ms
 2  10.39.1.6 (10.39.1.6)  [AS  32]  0.657 ms  0.611 ms  0.617 ms
 3  10.100.1.1 (10.100.1.1)  [AS  10, 40, 50]  0.880 ms  0.808 ms  0.774 ms
```

traceroute no-resolve

```
user@host> traceroute santacruz no-resolve
traceroute to host1.example.com (10.156.169.254), 30 hops max, 40 byte packets
 1  10.168.1.254  0.458 ms  0.370 ms  0.365 ms
 2  10.168.255.250  0.474 ms  0.450 ms  0.444 ms
 3  10.156.169.254  0.931 ms  0.876 ms  0.862 ms
```

traceroute propogate-ttl

```
user@host> traceroute propagate-ttl 100.200.2.2 routing-instance VPN-A
traceroute to 100.200.2.2 (100.200.2.2) from 1.1.0.2, 30 hops max, 40 byte packets
 1  1.2.0.2 (1.2.0.2)  2.456 ms  1.753 ms  1.672 ms
     MPLS Label=299776 CoS=0 TTL=1 S=0
     MPLS Label=299792 CoS=0 TTL=1 S=1
 2  1.3.0.2 (1.3.0.2)  1.213 ms  1.225 ms  1.166 ms
     MPLS Label=299792 CoS=0 TTL=1 S=1
 3  100.200.2.2 (100.200.2.2)  1.422 ms  1.521 ms  1.443 ms
```
tracert /r 5 10.168.2.224
tracert -h 5 10.168.2.224
tracert -d 10.168.2.224
tracert -a 10.168.2.224
tracert -w 5 10.168.2.224
tracert -t 10.168.2.224
tracert -n 5 10.168.2.224
tracert -f 5 10.168.2.224
tracert -j 5 10.168.2.224
tracert -k 5 10.168.2.224