



Junos[®] OS

Secure Neighbor Discovery Configuration Guide

Release

13.1



Published: 2013-02-14

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

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

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

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

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

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

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

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

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

Junos® OS Secure Neighbor Discovery Configuration Guide

13.1

Copyright © 2013, Juniper Networks, Inc.
All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

| | | |
|------------------|---|-----------|
| | About the Documentation | vii |
| | Documentation and Release Notes | vii |
| | Supported Platforms | vii |
| | Using the Examples in This Manual | vii |
| | Merging a Full Example | viii |
| | Merging a Snippet | viii |
| | Documentation Conventions | ix |
| | Documentation Feedback | xi |
| | Requesting Technical Support | xi |
| | Self-Help Online Tools and Resources | xi |
| | Opening a Case with JTAC | xii |
| Part 1 | Overview | |
| Chapter 1 | Introduction to Secure Neighbor Discovery Configuration Guidelines | 3 |
| | Secure Neighbor Discovery Configuration Overview | 3 |
| Part 2 | Configuration | |
| Chapter 2 | Concept and Example | 7 |
| | Example: Configuring Secure IPv6 Neighbor Discovery | 7 |
| | Understanding Secure IPv6 Neighbor Discovery | 7 |
| | Example: Configuring Secure IPv6 Neighbor Discovery | 7 |
| Chapter 3 | Configuration Statements | 11 |
| | cryptographic-address | 11 |
| | key-length | 12 |
| | key-pair | 12 |
| | neighbor-discovery | 13 |
| | secure | 14 |
| | security-level | 15 |
| | timestamp | 16 |
| | traceoptions (Protocols Secure Neighbor Discovery) | 17 |
| Part 3 | Administration | |
| Chapter 4 | Operational Commands | 21 |
| | monitor interface | 22 |
| | monitor start | 31 |
| | monitor stop | 33 |
| | ping | 34 |
| | show ipv6 neighbors | 38 |

| | | |
|------------------|--|-----------|
| | show ipv6 router-advertisement | 40 |
| | show log | 43 |
| | traceroute | 46 |
| Part 4 | Troubleshooting | |
| Chapter 5 | Routing Protocol Process Memory FAQs | 53 |
| | Routing Protocol Process Memory FAQs Overview | 53 |
| | Routing Protocol Process Memory FAQs | 54 |
| | Frequently Asked Questions: Routing Protocol Process Memory | 54 |
| | Frequently Asked Questions: Interpreting Routing Protocol Process-Related Command Outputs | 55 |
| | Frequently Asked Questions: Routing Protocol Process Memory Swapping | 58 |
| | Frequently Asked Questions: Troubleshooting the Routing Protocol Process | 59 |
| Part 5 | Index | |
| | Index | 63 |

List of Tables

| | | |
|------------------|--|------------|
| | About the Documentation | vii |
| | Table 1: Notice Icons | ix |
| | Table 2: Text and Syntax Conventions | ix |
| Part 3 | Administration | |
| Chapter 4 | Operational Commands | 21 |
| | Table 3: Output Control Keys for the monitor interface Command | 22 |
| | Table 4: Output Control Keys for the monitor interface traffic Command | 23 |
| | Table 5: monitor interface Output Fields | 24 |
| | Table 6: monitor start Output Fields | 31 |
| | Table 7: show ipv6 neighbors Output Fields | 38 |
| | Table 8: show ipv6 router-advertisement Output Fields | 40 |
| | Table 9: traceroute Output Fields | 48 |
| Part 4 | Troubleshooting | |
| Chapter 5 | Routing Protocol Process Memory FAQs | 53 |
| | Table 10: show system processes extensive Output Fields | 56 |
| | Table 11: show task memory Output Fields | 57 |

About the Documentation

- [Documentation and Release Notes on page vii](#)
- [Supported Platforms on page vii](#)
- [Using the Examples in This Manual on page vii](#)
- [Documentation Conventions on page ix](#)
- [Documentation Feedback on page xi](#)
- [Requesting Technical Support on page xi](#)

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 <http://www.juniper.net/techpubs/>.

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 <http://www.juniper.net/books>.

Supported Platforms

For the features described in this document, the following platforms are supported:

- [J Series](#)
- [SRX Series](#)
- [T Series](#)
- [MX Series](#)
- [M Series](#)

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.

```
system {
  scripts {
    commit {
      file ex-script.xsl;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

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.


```
commit {
  file ex-script-snippet.xml; }
```

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 the CLI User Guide.

Documentation Conventions

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

Table 1: Notice Icons




| Icon | Meaning | Description |
|---|--------------------|---|
|  | Informational note | Indicates important features or instructions. |
|  | Caution | Indicates a situation that might result in loss of data or hardware damage. |
|  | Warning | Alerts you to the risk of personal injury or death. |
|  | Laser warning | Alerts you to the risk of personal injury from a laser. |

Table 2 on page ix defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

| Convention | Description | Examples |
|----------------------------|--|--|
| Bold text like this | Represents text that you type. | To enter configuration mode, type the configure command: user@host> configure |
| Fixed-width text like this | Represents output that appears on the terminal screen. | user@host> show chassis alarms No alarms currently active |

Table 2: Text and Syntax Conventions (*continued*)

| Convention | Description | Examples |
|--------------------------------|--|---|
| <i>Italic text like this</i> | <ul style="list-style-type: none"> Introduces or emphasizes important new terms. Identifies book names. Identifies RFC and Internet draft titles. | <ul style="list-style-type: none"> A policy <i>term</i> is a named structure that defines match conditions and actions. <i>Junos OS System Basics Configuration Guide</i> RFC 1997, <i>BGP Communities Attribute</i> |
| <i>Italic text like this</i> | Represents variables (options for which you substitute a value) in commands or configuration statements. | Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i> |
| Text like this | Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components. | <ul style="list-style-type: none"> To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level. The console port is labeled CONSOLE. |
| < > (angle brackets) | Enclose optional keywords or variables. | stub <default-metric metric>; |
| (pipe symbol) | 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. | broadcast multicast <i>(string1 string2 string3)</i> |
| # (pound sign) | Indicates a comment specified on the same line as the configuration statement to which it applies. | rsvp { # Required for dynamic MPLS only |
| [] (square brackets) | Enclose a variable for which you can substitute one or more values. | community name members [community-ids] |
| Indentation and braces ({ }) | Identify a level in the configuration hierarchy. | [edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } } |
| ;(semicolon) | Identifies a leaf statement at a configuration hierarchy level. | |
| J-Web GUI Conventions | | |
| Bold text like this | Represents J-Web graphical user interface (GUI) items you click or select. | <ul style="list-style-type: none"> In the Logical Interfaces box, select All Interfaces. To cancel the configuration, click Cancel. |
| > (bold right angle bracket) | Separates levels in a hierarchy of J-Web selections. | In the configuration editor hierarchy, select Protocols>Ospf . |

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release 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 J-Care or JNASC 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.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at <http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.
- Product warranties—For product warranty information, visit <http://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: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>
- Download the latest versions of software and review release notes: <http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <https://www.juniper.net/alerts/>

- Join and participate in the Juniper Networks Community Forum:
<http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

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

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- 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 <http://www.juniper.net/support/requesting-support.html>.

PART 1

Overview

- [Introduction to Secure Neighbor Discovery Configuration Guidelines on page 3](#)

CHAPTER 1

Introduction to Secure Neighbor Discovery Configuration Guidelines

- [Secure Neighbor Discovery Configuration Overview on page 3](#)

Secure Neighbor Discovery Configuration Overview

The Secure Neighbor Discovery (SEND) Protocol provides support for protecting Neighbor Discovery Protocol (NDP) messages. SEND is applicable in environments where physical security on a link is not ensured and attacks on NDP messages are a concern. The Junos[®] operating system (Junos OS) implementation secures NDP messages through cryptographically generated addresses (CGAs).

You must also enable IPv6 on at least one interface. Because SEND relies on dynamically generated CGAs, it does not support static IPv6 addresses.

Related Documentation

- [Example: Configuring Secure IPv6 Neighbor Discovery on page 7](#)

PART 2

Configuration

- [Concept and Example on page 7](#)
- [Configuration Statements on page 11](#)

CHAPTER 2

Concept and Example

- [Example: Configuring Secure IPv6 Neighbor Discovery on page 7](#)

Example: Configuring Secure IPv6 Neighbor Discovery

- [Understanding Secure IPv6 Neighbor Discovery on page 7](#)
- [Example: Configuring Secure IPv6 Neighbor Discovery on page 7](#)

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.

Example: Configuring Secure IPv6 Neighbor Discovery

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

- [Requirements on page 7](#)
- [Overview on page 8](#)
- [Configuration on page 8](#)
- [Verification on page 10](#)

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 {
    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.

```
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.

```
[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.

```
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 10](#)
- [Tracing Neighbor Discovery Events on page 10](#)

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: recvd 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: recvd 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](#)

CHAPTER 3

Configuration Statements

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.

Required Privilege Level routing level—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 7](#)

key-length

| | |
|---------------------------------|---|
| Syntax | <code>key-length <i>number</i>;</code> |
| 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 | <i>number</i> —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 | <ul style="list-style-type: none">• Example: Configuring Secure IPv6 Neighbor Discovery on page 7 |

key-pair

| | |
|---------------------------------|---|
| Syntax | <code>key-pair <i>pathname</i>;</code> |
| 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 <code>/var/etc/rsa_key</code> directory. |
| Options | <i>pathname</i> —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 | <ul style="list-style-type: none">• Example: Configuring Secure IPv6 Neighbor Discovery on page 7 |

neighbor-discovery

```
Syntax  neighbor-discovery {
        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.

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 7](#)

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.

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 7](#)

security-level

| | |
|---------------------------------|--|
| Syntax | <code>security-level { (default secure-messages-only); }</code> |
| 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 | <ul style="list-style-type: none">• Example: Configuring Secure IPv6 Neighbor Discovery on page 7 |

timestamp

| | |
|---------------------------------|---|
| Syntax | <pre>timestamp { clock-drift <i>value</i>; known-peer-window <i>seconds</i>; new-peer-window <i>seconds</i>; }</pre> |
| 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 | <p>clock-drift <i>value</i>—Specify the allowable drift in time between the synchronization of peers. For <i>value</i>, specify a fractional value of 100.</p> <p>Default: 0.01</p> <p>known-peer-window <i>seconds</i>—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.</p> <p>Default: 1 second</p> <p>new-peer-window <i>seconds</i>—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.</p> <p>Default: 300 seconds</p> |
| Required Privilege Level | routing—To view this statement in the configuration. routing-control—To add this statement to the configuration. |
| Related Documentation | <ul style="list-style-type: none">• Example: Configuring Secure IPv6 Neighbor Discovery on page 7 |

traceoptions (Protocols Secure Neighbor Discovery)

| | |
|----------------------------|---|
| Syntax | <pre> traceoptions { file <i>filename</i> <files <i>number</i>> <match <i>regular-expression</i>> <size <i>size</i>> <world-readable no-world-readable>; flag <i>flag</i>; no-remote-trace; } </pre> |
| 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 | <p>file <i>filename</i>—Name of the file to receive the tracing operation. Enclose the name within quotation marks. All files are placed in the directory /var/log.</p> <p>files <i>number</i>—(Optional) Maximum number of trace files. When a trace file named <i>trace-file</i> reaches its maximum size, it is renamed <i>trace-file.0</i>, then <i>trace-file.1</i> and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.</p> <p>If you specify a maximum number of files, you must also specify a maximum file size with the size option.</p> <p>Range: 2 through 1000 files</p> <p>Default: 10 files</p> <p><i>flag</i>—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements.</p> <p>Secure Neighbor Discovery Tracing Options</p> <ul style="list-style-type: none"> • configuration—All configuration events. • cryptographic-address—Cryptographically generated address events. • protocol—All protocol processing events. • rsa—RSA events. <p>Global Tracing Options</p> <ul style="list-style-type: none"> • all—All tracing operations. <p>You can specify one or more of following flag modifiers:</p> <ul style="list-style-type: none"> • 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 | <ul style="list-style-type: none">• Example: Configuring Secure IPv6 Neighbor Discovery on page 7 |
|------------------------------|---|

PART 3

Administration

- [Operational Commands on page 21](#)

CHAPTER 4

Operational Commands

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.

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: This command is not supported on the QFX3000 QFabric system.

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 Junos OS Operational Mode Commands. To control the output of the **monitor interface** command while it is running, use the keys listed in [Table 3 on page 22](#). The keys are not case-sensitive.

Table 3: Output Control Keys for the monitor interface Command

| Key | Action |
|-----|--|
| c | 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. |
| f | Freezes the display, halting the display of updated statistics and delta counters. |
| i | Displays information about a different interface. The command prompts you for the name of a specific interface. |

Table 3: Output Control Keys for the monitor interface Command (*continued*)

| Key | Action |
|----------|--|
| n | 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. |
| q or Esc | Quits the command and returns to the command prompt. |
| t | Thaws the display, resuming the update of the statistics and delta counters. |

To control the output of the **monitor interface traffic** command while it is running, use the keys listed in [Table 4 on page 23](#). The keys are not case-sensitive.

Table 4: Output Control Keys for the monitor interface traffic Command

| Key | Action |
|----------|--|
| b | Displays the statistics in units of bytes and bytes per second (Bps). |
| c | Clears (return to 0) the delta counters in the Current Delta column. The statistics counters are not cleared. |
| d | Displays the Current Delta column (instead of the rate column) in Bps or packets per second (pps). |
| p | Displays the statistics in units of packets and packets per second (pps). |
| q or Esc | Quits the command and returns to the command prompt. |
| r | Displays the rate column (instead of the Current Delta column) in Bps and pps. |

Required Privilege Level trace

List of Sample Output [monitor interface \(Physical\) on page 26](#)
[monitor interface \(OTN Interface\) on page 26](#)
[monitor interface \(Logical\) on page 27](#)
[monitor interface \(QFX3500 Switch\) on page 28](#)
[monitor interface traffic on page 28](#)
[monitor interface traffic \(QFX3500 Switch\) on page 29](#)
[monitor interface traffic detail \(QFX3500 Switch\) on page 29](#)

Output Fields [Table 5 on page 24](#) describes the output fields for the **monitor interface** command. Output fields are listed in the approximate order in which they appear.

Table 5: monitor interface Output Fields

| Field Name | Field Description | Level of Output |
|--------------------------|--|-----------------|
| routerl | Hostname of the router. | All levels |
| Seconds | How long the monitor interface command has been running or how long since you last cleared the counters. | All levels |
| Time | Current time (UTC). | All levels |
| Delay x/y/z | Time difference between when the statistics were displayed and the actual clock time. <ul style="list-style-type: none"> • x—Time taken for the last polling (in milliseconds). • y—Minimum time taken across all pollings (in milliseconds). • z—Maximum time taken across all pollings (in milliseconds). | All levels |
| Interface | Short description of the interface, including its name, status, and encapsulation. | All levels |
| Link | State of the link: Up , Down , or Test . | All levels |
| Current delta | 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. | All levels |
| Local Statistics | (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. <ul style="list-style-type: none"> • Input bytes—Number of bytes received on the interface. • Output bytes—Number of bytes transmitted on the interface. • Input packets—Number of packets received on the interface. • Output packets—Number of packets transmitted on the interface. | All levels |
| Remote Statistics | (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. <ul style="list-style-type: none"> • Input bytes—Number of bytes received on the interface. • Output bytes—Number of bytes transmitted on the interface. • Input packets—Number of packets received on the interface. • Output packets—Number of packets transmitted on the interface. | All levels |

Table 5: monitor interface Output Fields (*continued*)

| Field Name | Field Description | Level of Output |
|--------------------|--|-----------------|
| Traffic statistics | <p>Total number of bytes and packets received and transmitted on the interface. These statistics are the sum of the local and remote statistics. 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.</p> <ul style="list-style-type: none"> • Input bytes—Number of bytes received on the interface. • Output bytes—Number of bytes transmitted on the interface. • Input packets—Number of packets received on the interface. • Output packets—Number of packets transmitted on the interface. | All levels |
| Description | With the traffic option, displays the interface description configured at the [edit interfaces <i>interface-name</i>] hierarchy level. | detail |

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: 0C48
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]
Output drops:                        0                               [0]
Aged packets:                        0                               [0]
Active alarms : None
Active defects: None
SONET error counts/seconds:
LOS count                            1                               [0]
LOF count                            1                               [0]
SEF count                            1                               [0]
ES-S                                 0                               [0]
SES-S                                 0                               [0]
SONET statistics:
BIP-B1                               458871                          [0]
BIP-B2                               460072                          [0]
REI-L                                465610                          [0]
BIP-B3                               458978                          [0]
REI-P                                458773                          [0]
Received SONET overhead:
F1      : 0x00 J0      : 0x00 K1      : 0x00
K2      : 0x00 S1      : 0x00 C2      : 0x00
C2(cmp) : 0x00 F2      : 0x00 Z3      : 0x00
Z4      : 0x00 S1(cmp) : 0x00
Transmitted SONET overhead:
F1      : 0x00 J0      : 0x01 K1      : 0x00
K2      : 0x00 S1      : 0x00 C2      : 0xcf
F2      : 0x00 Z3      : 0x00 Z4      : 0x00

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

```

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
(Logical)

user@host> 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
Encapsulation: PPP
Local statistics:
    Input bytes: 0
    Output bytes: 0
    Input packets: 0
    Output packets: 0
Remote statistics:
    Input bytes: 0 (0 bps)
    Output bytes: 0 (0 bps)
    Input packets: 0 (0 pps)
    Output packets: 0 (0 pps)
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)

```

user@switch> monitor interface ge-0/0/0
Interface: ge-0/0/0, Enabled, Link is Down
Encapsulation: Ethernet, Speed: Unspecified
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: 0
    Output errors: 0
    Output drops: 0
    Aged packets: 0
Active alarms : LINK
Active defects: LINK
Input MAC/Filter statistics:
    Unicast packets 0
    Broadcast packets 0 Multicast packet
Interface warnings:
    o Outstanding LINK alarm

```

monitor interface traffic

```

user@host> monitor interface traffic
host name          Seconds: 15          Time: 12:31:09

Interface  Link  Input packets  (pps)  Output packets  (pps)
so-1/0/0   Down    0             (0)      0             (0)
so-1/1/0   Down    0             (0)      0             (0)
so-1/1/1   Down    0             (0)      0             (0)
so-1/1/2   Down    0             (0)      0             (0)
so-1/1/3   Down    0             (0)      0             (0)
t3-1/2/0   Down    0             (0)      0             (0)
t3-1/2/1   Down    0             (0)      0             (0)
t3-1/2/2   Down    0             (0)      0             (0)
t3-1/2/3   Down    0             (0)      0             (0)

```


| | | | | | |
|----------|------|--------|-----|-------|-----|
| so-2/0/0 | Up | 211035 | (1) | 36778 | (0) |
| so-2/0/1 | Up | 192753 | (1) | 36782 | (0) |
| so-2/0/2 | Up | 211020 | (1) | 36779 | (0) |
| so-2/0/3 | Up | 211029 | (1) | 36776 | (0) |
| so-2/1/0 | Up | 189378 | (1) | 36349 | (0) |
| so-2/1/1 | Down | 0 | (0) | 18747 | (0) |
| so-2/1/2 | Down | 0 | (0) | 16078 | (0) |
| so-2/1/3 | Up | 0 | (0) | 80338 | (0) |
| at-2/3/0 | Up | 0 | (0) | 0 | (0) |
| at-2/3/1 | Down | 0 | (0) | 0 | (0) |

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
```

| Interface | Link | Input packets | (pps) | Output packets | (pps) |
|-----------|------|---------------|-------|----------------|-------|
| ge-0/0/0 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/1 | Up | 392187 | (0) | 392170 | (0) |
| ge-0/0/2 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/3 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/4 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/5 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/6 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/7 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/8 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/9 | Up | 392184 | (0) | 392171 | (0) |
| ge-0/0/10 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/11 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/12 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/13 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/14 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/15 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/16 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/17 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/18 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/19 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/20 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/21 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/22 | Up | 392172 | (0) | 392187 | (0) |
| ge-0/0/23 | Up | 392185 | (0) | 392173 | (0) |
| vcp-0 | Down | 0 | | 0 | |
| vcp-1 | Down | 0 | | 0 | |
| ae0 | Down | 0 | (0) | 0 | (0) |
| bme0 | Up | 0 | | 1568706 | |

monitor interface traffic detail (QFX3500 Switch)

```
user@switch> monitor interface traffic detail
switch                                     Seconds: 74                               Time: 16:03:02
```

| Interface | Link | Input packets | (pps) | Output packets | (pps) |
|-----------|------|---------------|-------|----------------|-------|
| ge-0/0/0 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/1 | Up | 392183 | (0) | 392166 | (0) |
| ge-0/0/2 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/3 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/4 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/5 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/6 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/7 | Down | 0 | (0) | 0 | (0) |

| | | | | | |
|-----------|------|--------|-----|---------|-----|
| ge-0/0/8 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/9 | Up | 392181 | (0) | 392168 | (0) |
| ge-0/0/10 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/11 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/12 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/13 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/14 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/15 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/16 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/17 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/18 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/19 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/20 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/21 | Down | 0 | (0) | 0 | (0) |
| ge-0/0/22 | Up | 392169 | (0) | 392184 | (1) |
| ge-0/0/23 | Up | 392182 | (0) | 392170 | (0) |
| vcp-0 | Down | 0 | | 0 | |
| vcp-1 | Down | 0 | | 0 | |
| ae0 | Down | 0 | (0) | 0 | (0) |
| bme0 | Up | 0 | | 1568693 | |

monitor start

| | |
|-------------------------------|--|
| Syntax | <code>monitor start <i>filename</i></code> |
| 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 | <i>filename</i> —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 Level | trace |
| Related Documentation | <ul style="list-style-type: none"> monitor list monitor stop on page 33 |
| List of Sample Output | monitor start on page 32 |
| Output Fields | Table 6 on page 31 describes the output fields for the monitor start command. Output fields are listed in the approximate order in which they appear. |

Table 6: monitor start Output Fields

| Field Name | Field Description |
|------------------------------|--|
| ***<i>filename</i>*** | Name of the file from which entries are being displayed. This line is displayed initially and when the command switches between log files. |
| <i>Date and time</i> | Timestamp for the log entry. |

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: RSA key generation complete.
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: RSA authentication for root accepted.
Jul 20 15:07:37 hang sshd[5845]: log: ROOT LOGIN as 'root' from trip.jcmax.com
Jul 20 15:07:37 hang sshd[5845]: log: Closing connection to 204.69.248.180
```

monitor stop

| | |
|---------------------------------|---|
| Syntax | <code>monitor stop <i>filename</i></code> |
| 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 | <i>filename</i> —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 <i>protocol</i>] hierarchy levels. |
| Required Privilege Level | trace |
| Related Documentation | <ul style="list-style-type: none"> monitor list monitor start on page 31 |
| List of Sample Output | monitor stop on page 33 |
| Output Fields | This command produces no output. |

Sample Output

`monitor stop` user@host> `monitor stop`

ping

Syntax `ping host`
 `<bypass-routing>`
 `<count requests>`
 `<detail>`
 `<do-not-fragment>`
 `<inet | inet6>`
 `<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>`
 `<vpls instance-name>`
 `<wait seconds>`

Syntax (QFX Series) `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>`

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.

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.

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) 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 IPv6 packets, this option disables fragmentation.



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.

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, enter the **clear cli logical-system** command.

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

mac-address *mac-address*—(Optional) Ping the physical or hardware address of the remote system you are trying to reach.

no-resolve—(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.

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**.

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

- [Configuring the Junos OS ICMPv4 Rate Limit for ICMPv4 Routing Engine Messages](#)

List of Sample Output

[ping hostname on page 37](#)
[ping hostname rapid on page 37](#)

[ping hostname size count on page 37](#)

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.

Sample Output**ping hostname**

```
user@host> ping skye
PING skye.net (192.168.169.254): 56 data bytes
64 bytes from 192.168.169.254: icmp_seq=0 ttl=253 time=1.028 ms
64 bytes from 192.168.169.254: icmp_seq=1 ttl=253 time=1.053 ms
64 bytes from 192.168.169.254: icmp_seq=2 ttl=253 time=1.025 ms
64 bytes from 192.168.169.254: icmp_seq=3 ttl=253 time=1.098 ms
64 bytes from 192.168.169.254: icmp_seq=4 ttl=253 time=1.032 ms
64 bytes from 192.168.169.254: icmp_seq=5 ttl=253 time=1.044 ms
^C [abort]
```

ping hostname rapid

```
user@host> ping skye rapid
PING skye.net (192.168.169.254): 56 data bytes
!!!!!!
--- skye.net 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 skye size 200 count 5
PING skye.net (192.168.169.254): 200 data bytes
208 bytes from 192.168.169.254: icmp_seq=0 ttl=253 time=1.759 ms
208 bytes from 192.168.169.254: icmp_seq=1 ttl=253 time=2.075 ms
208 bytes from 192.168.169.254: icmp_seq=2 ttl=253 time=1.843 ms
208 bytes from 192.168.169.254: icmp_seq=3 ttl=253 time=1.803 ms
208 bytes from 192.168.169.254: icmp_seq=4 ttl=253 time=17.898 ms

--- skye.net ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.759/5.075/17.898 ms
```

show ipv6 neighbors

| | |
|---------------------------------|--|
| Syntax | show ipv6 neighbors |
| 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. |
| Description | Display information about the IPv6 neighbor cache. |
| Options | This command has no options. |
| Required Privilege Level | view |
| Related Documentation | <ul style="list-style-type: none"> clear ipv6 neighbors |
| List of Sample Output | show ipv6 neighbors on page 39 |
| Output Fields | Table 7 on page 38 describes the output fields for the show ipv6 neighbors command. Output fields are listed in the approximate order in which they appear. |

Table 7: show ipv6 neighbors Output Fields

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

Sample Output

show ipv6 neighbors

user@host> show ipv6 neighbors

| IPv6 Address | Linklayer Address | State | Exp | Rtr | Secure |
|---|-------------------|-------|-----|-----|--------|
| Interface | | | | | |
| 2001:db8:0:1:2a0:a514:0:24c fe-1/2/0.1 | 00:05:85:8f:c8:bd | stale | 546 | yes | no |
| fe80::2a0:a514:0:24c fe-1/2/0.1 | 00:05:85:8f:c8:bd | stale | 258 | yes | no |
| fe80::2a0:a514:0:64c fe-1/2/1.5 | 00:05:85:8f:c8:bd | stale | 111 | yes | no |
| fe80::2a0:a514:0:a4c fe-1/2/2.9 | 00:05:85:8f:c8:bd | stale | 327 | yes | no |

show ipv6 router-advertisement

| | |
|---------------------------------|--|
| Syntax | <pre>show ipv6 router-advertisement <conflicts> <interface <i>interface</i>> <logical-system (all <i>logical-system-name</i>)> <prefix <i>prefix/prefix length</i>></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 12.2 for the QFX Series.</p> |
| 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 | <p>none—Display all IPv6 router advertisement information for all interfaces.</p> <p>conflicts—(Optional) Display only the IPv6 router advertisement information that is conflicting.</p> <p>interface <i>interface</i>—(Optional) Display IPv6 router advertisement information for the specified interface.</p> <p>logical-system (all <i>logical-system-name</i>)—(Optional) Perform this operation on all logical systems or on a particular logical system.</p> <p>prefix <i>prefix/prefix length</i>—(Optional) Display IPv6 router advertisement information for the specified prefix.</p> |
| Additional Information | The display identifies conflicting information by enclosing the value the router is advertising in brackets. |
| Required Privilege Level | view |
| Related Documentation | <ul style="list-style-type: none"> clear ipv6 router-advertisement |
| List of Sample Output | show ipv6 router-advertisement on page 42 show ipv6 router-advertisement conflicts on page 42 show ipv6 router-advertisement prefix on page 42 |
| Output Fields | Table 8 on page 40 describes the output fields for the show ipv6 router-advertisement command. Output fields are listed in the approximate order in which they appear. |

Table 8: show ipv6 router-advertisement Output Fields

| Field Name | Field Description |
|---------------------|---|
| Interface | Name of the interface. |
| Advertisements sent | Number of router advertisements sent and the elapsed time since they were sent. |

Table 8: show ipv6 router-advertisement Output Fields (*continued*)

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

Sample Output

**show ipv6
router-advertisement**

```
user@host> show ipv6 router-advertisement
Interface: fe-0/1/1.0
  Advertisements sent: 0
  Solicits received: 0
  Advertisements 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 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: 8040:1::/64
    Valid lifetime: 2592000 sec
    Preferred lifetime: 604800 sec
    On link: 1
    Autonomous: 1
```

show log

| | |
|-----------------------------------|---|
| Syntax | show log <filename user <username>> |
| Syntax (QFabric System) | show log <filename> |
| Syntax (TX Matrix Routers) | show log <all-lcc lcc <i>number</i> 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. |
| Description | List log files, display log file contents, or display information about users who have logged in to the router or switch. |
| Options | <p>none—List all log files.</p> <p><all-lcc lcc <i>number</i> scc>—(TX Matrix routers only) (Optional) Display logging information about all T640 routers (or line-card chassis) or a specific T640 router (replace <i>number</i> 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).</p> <p>filename—(Optional) Display the log messages in the specified log file. For the routing matrix, the filename must include the chassis information.</p> <p>user <username>—(Optional) Display logging information about users who have recently logged in to the router or switch. If you include <i>username</i>, display logging information about the specified user.</p> |
| Required Privilege Level | trace |
| List of Sample Output | show log on page 44 show log filename on page 44 show log filename (QFabric System) on page 44 show log user on page 45 |

Sample Output

show log

```
user@host> show log
total 57518
-rw-r--r-- 1 root bin      211663 Oct  1 19:44 dcd
-rw-r--r-- 1 root bin      999947 Oct  1 19:41 dcd.0
-rw-r--r-- 1 root bin      999994 Oct  1 17:48 dcd.1
-rw-r--r-- 1 root bin      238815 Oct  1 19:44 rpd
-rw-r--r-- 1 root bin     1049098 Oct  1 18:00 rpd.0
-rw-r--r-- 1 root bin     1061095 Oct  1 12:13 rpd.1
-rw-r--r-- 1 root bin     1052026 Oct  1 06:08 rpd.2
-rw-r--r-- 1 root bin     1056309 Sep 30 18:21 rpd.3
-rw-r--r-- 1 root bin     1056371 Sep 30 14:36 rpd.4
-rw-r--r-- 1 root bin     1056301 Sep 30 10:50 rpd.5
-rw-r--r-- 1 root bin     1056350 Sep 30 07:04 rpd.6
-rw-r--r-- 1 root bin     1048876 Sep 30 03:21 rpd.7
-rw-rw-r-- 1 root bin       19656 Oct  1 19:37 wtmp
```

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 rcv len 56 V9 seq 148 op add Type route/if af 2 addr
13.13.13.21 nhop type local nhop 13.13.13.21
Oct  1 18:00:19 KRT rcv len 56 V9 seq 149 op add Type route/if af 2 addr
13.13.13.22 nhop type unicast nhop 13.13.13.22
Oct  1 18:00:19 KRT rcv len 48 V9 seq 150 op add Type ifaddr index 24 devindex
43
Oct  1 18:00:19 KRT rcv len 144 V9 seq 151 op chnge Type ifdev devindex 44
Oct  1 18:00:19 KRT rcv len 144 V9 seq 152 op chnge Type ifdev devindex 45
Oct  1 18:00:19 KRT rcv len 144 V9 seq 153 op chnge Type ifdev devindex 46
Oct  1 18:00:19 KRT rcv len 1272 V9 seq 154 op chnge Type ifdev devindex 47
...
```

show log filename (QFabric System)

```
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 2, 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)
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 2, jnxFruL3Index 0,
jnxFruName PIC: @ 0/1/*, jnxFruType 11, jnxFruSlot 0, jnxFruOfflineReason 2,
jnxFruLastPowerOff 0, jnxFruLastPowerOn 242757)
Mar 28 18:00:16 qfabric file: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:16 ED1486
file: UI_COMMIT: User 'root' requested 'commit' operation (comment: none)
Mar 28 18:00:27 qfabric file: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:27 ED1486
```



```

file: UI_COMMIT: User 'root' requested 'commit' operation (comment: none)
Mar 28 18:00:50 qfabric file: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:50
_DCF_default__NW-INE-0_RE0_ file: UI_COMMIT: User 'root' requested 'commit'
operation (comment: none)
Mar 28 18:00:50 qfabric file: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:50
_DCF_default__NW-INE-0_RE0_ file: UI_COMMIT: User 'root' requested 'commit'
operation (comment: none)
Mar 28 18:00:55 qfabric file: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:00:55 ED1492
file: UI_COMMIT: User 'root' requested 'commit' operation (comment: none)
Mar 28 18:01:10 qfabric file: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:01:10 ED1492
file: UI_COMMIT: User 'root' requested 'commit' operation (comment: none)
Mar 28 18:02:37 qfabric chassisd: QFABRIC_INTERNAL_SYSLOG: Mar 28 18:02:37 ED1491
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 33809)

```

show log user

```

user@host> show log user
darius  mg2546                Thu Oct  1 19:37   still logged in
darius  mg2529                Thu Oct  1 19:08 - 19:36   (00:28)
darius  mg2518                Thu Oct  1 18:53 - 18:58   (00:04)
root    mg1575                Wed Sep 30 18:39 - 18:41   (00:02)
root    ttyp2      jun.site.per    Wed Sep 30 18:39 - 18:41   (00:02)
alex    ttyp1      192.168.1.2    Wed Sep 30 01:03 - 01:22   (00:19)

```

traceroute

| | |
|----------------------------|---|
| Syntax | <pre>traceroute <i>host</i> <as-number-lookup> <bypass-routing> <clns> <gateway <i>address</i>> <inet inet6> <interface <i>interface-name</i>> <logical system <i>logical-system-name</i>> <monitor <i>host</i>> <mpls (<i>ldp FEC address</i> <i>rsvp label-switched-path-name</i>)> <no-resolve> <propagate-ttl> <routing-instance <i>routing-instance-name</i>> <source <i>source-address</i>> <tos <i>value</i>> <ttl <i>value</i>> <wait <i>seconds</i>></pre> |
| Syntax (QFX Series) | <pre>traceroute <i>host</i> <as-number-lookup> <bypass-routing> <gateway <i>address</i>> <inet> <interface <i>interface-name</i>> <monitor <i>host</i>> <no-resolve> <routing-instance <i>routing-instance-name</i>> <source <i>source-address</i>> <tos <i>value</i>> <ttl <i>value</i>> <wait <i>seconds</i>></pre> |
| Release Information | <p>Command introduced before Junos OS Release 7.4.</p> <p>Command introduced in Junos OS Release 9.0 for EX Series switches.</p> <p>mpls option introduced in Junos OS Release 9.2.</p> <p>Command introduced in Junos OS Release 11.1 for the QFX Series.</p> <p>propagate-ttl option introduced in Junos OS Release 12.1.</p> |
| 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 | <p>host—IP address or name of remote host.</p> <p>as-number-lookup—(Optional) Display the autonomous system (AS) number of each intermediate hop on the path from the host to the destination.</p> <p>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.</p> |

clns—(Optional) Trace the route belonging to the 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 logical-system-name—(Optional) 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.

mpls (ldp FEC address | rsvp label-switched-path name)—(Optional) See **traceroute mpls ldp** and **traceroute mpls rsvp**.

no-resolve—(Optional) Do not attempt to determine the hostname that corresponds to the IP address.

propagate-ttl—(Optional) On the PE router, 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-propagate-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 49](#)
[traceroute as-number-lookup host on page 49](#)

[traceroute no-resolve on page 49](#)

[traceroute propagate-ttl on page 49](#)

[traceroute \(Between CE Routers, Layer 3 VPN\) on page 49](#)

[traceroute \(Through an MPLS LSP\) on page 49](#)

Output Fields [Table 9 on page 48](#) describes the output fields for the **traceroute** command. Output fields are listed in the approximate order in which they appear.

Table 9: traceroute Output Fields

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

Sample Output

traceroute

```
user@host> traceroute santacruz
traceroute to green.company.net (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 green.company.net (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 propagate-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
```

traceroute (Between CE Routers, Layer 3 VPN)

```
user@host> traceroute vpn09
traceroute to vpn09.skybank.net (10.255.14.179), 30 hops max, 40
byte packets
 1 10.39.10.21 (10.39.10.21) 0.598 ms 0.500 ms 0.461 ms
 2 10.39.1.13 (10.39.1.13) 0.796 ms 0.775 ms 0.806 ms
   MPLS Label=100006 CoS=0 TTL=1 S=1
 3 vpn09.skybank.net (10.255.14.179) 0.783 ms 0.716 ms 0.686
```

traceroute (Through an MPLS LSP)

```
user@host> traceroute mpls1
traceroute to 10.168.1.224 (10.168.1.224), 30 hops max, 40 byte packets
 1 mpls1-sr0.company.net (10.168.200.101) 0.555 ms 0.393 ms 0.367 ms
   MPLS Label=1024 CoS=0 TTL=1
 2 mpls5-lo0.company.net (10.168.1.224) 0.420 ms 0.394 ms 0.401 ms
```


PART 4

Troubleshooting

- [Routing Protocol Process Memory FAQs on page 53](#)

CHAPTER 5

Routing Protocol Process Memory FAQs

- [Routing Protocol Process Memory FAQs Overview on page 53](#)
- [Routing Protocol Process Memory FAQs on page 54](#)

Routing Protocol Process Memory FAQs Overview

Junos OS is based on the FreeBSD Unix operating system. The open source software is modified and hardened to operate in the device's specialized environment. For example, some executables have been deleted, while other utilities were de-emphasized. Additionally, certain software processes were added to enhance the routing functionality. The result of this transformation is the kernel, the heart of the Junos OS software.

The kernel is responsible for operating multiple processes that perform the actual functions of the device. Each process operates in its own protected memory space, while the communication among all the processes is still controlled by the kernel. This separation provides isolation between the processes, and resiliency in the event of a process failure. This is important in a core routing platform because a single process failure does not cause the entire device to cease functioning.

Some of the common software processes include the routing protocol process (rpd) that controls the device's protocols, the device control process (dcd) that controls the device's interfaces, the management process (mgd) that controls user access to the device, the chassis process (chassisd) that controls the device's properties itself, and the Packet Forwarding Engine process (pfed) that controls the communication between the device's Packet Forwarding Engine and the Routing Engine. The kernel also generates specialized processes as needed for additional functionality, such as SNMP, the Virtual Router Redundancy Protocol (VRRP), and Class of Service (CoS).

The routing protocol process is a software process within the Routing Engine software, which controls the routing protocols that run on the device. Its functionality includes all protocol messages, routing table updates, and implementation of routing policies.

The routing protocol process starts all configured routing protocols and handles all routing messages. It maintains one or more routing tables, which consolidate the routing information learned from all routing protocols. From this routing information, the routing protocol process determines the active routes to network destinations and installs these routes into the Routing Engine's forwarding table. Finally, it implements routing policy, which allows you to control the routing information that is transferred between the routing

protocols and the routing table. Using routing policy, you can filter and limit the transfer of information as well as set properties associated with specific routes.

Related Documentation

- [Routing Protocol Process Memory FAQs on page 54](#)

Routing Protocol Process Memory FAQs

The following sections present the most frequently asked questions and answers related to the routing protocol process memory utilization, operation, interpretation of related command outputs, and troubleshooting the software process.

Frequently Asked Questions: Routing Protocol Process Memory

This section presents frequently asked questions and answers related to the memory usage of the routing protocol process.

Why does the routing protocol process use excessive memory?

The routing protocol process uses hundreds of megabytes of RAM in the Routing Engine to store information needed for the operation of routing and related protocols, such as BGP, OSPF, IS-IS, RSVP, LDP and MPLS. Such huge consumption of memory is common for the process, as the information it stores includes routes, next hops, interfaces, routing policies, labels, and label-switched paths (LSPs). Because access to the RAM memory is much faster than access to the hard disk, most of the routing protocol process information is stored in the RAM memory instead of using the hard disk space. This ensures that the performance of the routing protocol process is maximized.

How can I check the amount of memory the routing protocol process is using?

You can check routing protocol process memory usage by entering the **show system processes** and the **show task memory** Junos OS command-line interface (CLI) operational mode commands.

The **show system processes** command displays information about software processes that are running on the device and that have controlling terminals. The **show task memory** command displays memory utilization for routing protocol tasks on the Routing Engine.

You can check the routing protocol process memory usage by using the **show system processes** command with the **extensive** option. The **show task memory** command displays a report generated by the routing protocol process on its own memory usage. However, this report does not display all the memory used by the process. The value reported by the routing protocol process does not account for the memory used for the **TEXT** and **STACK** segments, or the memory used by the process's internal memory manager. Further, the Resident Set Size value includes shared library pages used by the routing protocol process.

For more information about checking the routing protocol process memory usage, see [Check Routing Protocol Process \(rpd\) Memory Usage](#).

For more information, see the **show system processes** command and the **show task memory** command.

I just deleted a large number of routes from the routing protocol process. Why is it still using so much memory?

The **show system processes extensive** command displays a **RES** value measured in kilobytes. This value represents the amount of program memory resident in the physical memory. This is also known as RSS or Resident Set Size. The **RES** value includes shared library pages used by the process. Any amount of memory freed by the process might still be considered part of the **RES** value. Generally, the kernel delays the migrating of memory out of the **Inact** queue into the **Cache** or **Free** list unless there is a memory shortage. This can lead to large discrepancies between the values reported by the routing protocol process and the kernel, even after the routing protocol process has freed a large amount of memory.

Frequently Asked Questions: Interpreting Routing Protocol Process-Related Command Outputs

This section presents frequently asked questions and answers about the routing protocol process-related Junos OS command-line interface (CLI) command outputs that are used to display the memory usage of the routing protocol process.

How do I interpret memory numbers displayed in the show system processes extensive command output?

The **show system processes extensive** command displays exhaustive system process information about software processes that are running on the device and have controlling terminals. This command is equivalent to the UNIX **top** command. However, the UNIX **top** command shows real-time memory usage, with the memory values constantly changing, while the **show system processes extensive** command provides a snapshot of memory usage in a given moment.

To check overall CPU and memory usage, enter the **show system processes extensive** command. Refer to [Table 10 on page 56](#) for information about the **show system processes extensive** commands output fields.

```
user@host> show system processes extensive
last pid: 544; load averages: 0.00, 0.00, 0.00 18:30:33
37 processes: 1 running, 36 sleeping

Mem: 25M Active, 3968K Inact, 19M Wired, 184K Cache, 8346K Buf, 202M Free
Swap: 528M Total, 64K Used, 528M Free

  PID USERNAME PRI NICE SIZE RES STATE TIME WCPU CPU COMMAND
    544 root    30  0  604K 768K RUN   0:00 0.00% 0.00% top
      3 root    28  0    0K  12K psleep 0:00 0.00% 0.00% vmdaemon
      4 root    28  0    0K  12K update 0:03 0.00% 0.00% update
    528 aviva    18  0  660K 948K pause  0:00 0.00% 0.00% tcsh
    204 root    18  0  300K 544K pause  0:00 0.00% 0.00% csh
    131 root    18  0  332K 532K pause  0:00 0.00% 0.00% cron
    186 root    18  0  196K  68K pause  0:00 0.00% 0.00% watchdog
     27 root    10  0  512M 16288K mfsidl 0:00 0.00% 0.00% mount_mfs
      1 root    10  0  620K 344K wait   0:00 0.00% 0.00% init
    304 root     3  0  884K 900K ttyin  0:00 0.00% 0.00% bash
    200 root     3  0  180K 540K ttyin  0:00 0.00% 0.00% getty
    203 root     3  0  180K 540K ttyin  0:00 0.00% 0.00% getty
    202 root     3  0  180K 540K ttyin  0:00 0.00% 0.00% getty
    201 root     3  0  180K 540K ttyin  0:00 0.00% 0.00% getty
    194 root     2  0 2248K 1640K select 0:11 0.00% 0.00% rpd
    205 root     2  0   964K  800K select 0:12 0.00% 0.00% tnp.chassisd
```

```

189 root      2  -12   352K   740K select  0:03  0.00%  0.00% xntpd
114 root      2   0   296K   612K select  0:00  0.00%  0.00% amd
188 root      2   0   780K   600K select  0:00  0.00%  0.00% dcd
527 root      2   0   176K   580K select  0:00  0.00%  0.00% rlogind
195 root      2   0   212K   552K select  0:00  0.00%  0.00% inetd
187 root      2   0   192K   532K select  0:00  0.00%  0.00% tnetd
 83 root      2   0   188K   520K select  0:00  0.00%  0.00% syslogd
538 root      2   0  1324K   516K select  0:00  0.00%  0.00% mgd
 99 daemon    2   0   176K   492K select  0:00  0.00%  0.00% portmap
163 root      2   0   572K   420K select  0:00  0.00%  0.00% nsrexecd
192 root      2   0   560K   400K select  0:10  0.00%  0.00% snmpd
191 root      2   0  1284K   376K select  0:00  0.00%  0.00% mgd
537 aviva     2   0   636K   364K select  0:00  0.00%  0.00% cli
193 root      2   0   312K   204K select  0:07  0.00%  0.00% mib2d
  5 root      2   0     0K    12K pfesel  0:00  0.00%  0.00% if_pfe
  2 root     -18   0     0K    12K psleep  0:00  0.00%  0.00% pagedaemon
  0 root     -18   0     0K     0K sched   0:00  0.00%  0.00% swapper

```

Table 10 on page 56 describes the output fields that represent the memory values for the **show system processes extensive** command. Output fields are listed in the approximate order in which they appear.

Table 10: show system processes extensive Output Fields

| Field Name | Field Description |
|---------------|---|
| Mem | Information about physical and virtual memory allocation. |
| Active | Memory allocated and actively used by the program. |
| Inact | Memory allocated but not recently used or memory freed by the programs. Inactive memory remains mapped in the address space of one or more processes and, therefore, counts toward the RSS value of those processes. |
| Wired | Memory that is not eligible to be swapped, usually used for in-kernel memory structures and/or memory physically locked by a process. |
| Cache | Memory that is not associated with any program and does not need to be swapped before being reused. |
| Buf | Size of memory buffer used to hold data recently called from the disk. |
| Free | Memory that is not associated with any programs. Memory freed by a process can become Inactive , Cache , or Free , depending on the method used by the process to free the memory. |
| Swap | Information about swap memory. <ul style="list-style-type: none"> • Total—Total memory available to be swapped to disk. • Used—Memory swapped to disk. • Free—Memory available for further swap. |

The rest of the command output displays information about the memory usage of each process. The **SIZE** field indicates the size of the virtual address space, and the **RES** field indicates the amount of the program in physical memory, which is also known as RSS or Resident Set Size. For more information, see the **show system processes** command.

What is the difference between Active and Inact memory that is displayed by the show system processes extensive command?

When the system is under memory pressure, the pageout process reuses memory from the free, cache, inactive and, if necessary, active pages. When the pageout process runs, it scans memory to see which pages are good candidates to be unmapped and freed up. Thus, the distinction between **Active** and **Inact** memory is only used by the pageout process to determine which pool of pages to free first at the time of a memory shortage.

The pageout process first scans the **Inact** list, and checks whether the pages on this list have been accessed since the time they have been listed here. The pages that have been accessed are moved from the **Inact** list to the **Active** list. On the other hand, pages that have not been accessed become prime candidates to be freed by the pageout process. If the pageout process cannot produce enough free pages from the **Inact** list, pages from the **Active** list get freed up.

Because the pageout process runs only when the system is under memory pressure, the pages on the **Inact** list remain untouched – even if they have not been accessed recently – when the amount of **Free** memory is adequate.

How do I interpret memory numbers displayed in the show task memory command output?

The **show task memory** command provides a comprehensive picture of the memory utilization for routing protocol tasks on the Routing Engine. The routing protocol process is the main task that uses Routing Engine memory.

To check routing process memory usage, enter the **show task memory** command. Refer to [Table 11 on page 57](#) for information about the **show task memory** command output fields.

```
user@host> show task memory
Memory          Size (kB)  %Available  When
Currently In Use:    29417      3%         now
Maximum Ever Used:   33882      4%         00/02/11 22:07:03
Available:          756281    100%        now
```

[Table 11 on page 57](#) describes the output fields for the **show task memory** command. Output fields are listed in the approximate order in which they appear.

Table 11: show task memory Output Fields

| Field Name | Field Description |
|--------------------------|---|
| Memory Currently In Use | Memory currently in use. Dynamically allocated memory plus the DATA segment memory in kilobytes. |
| Memory Maximum Ever Used | Maximum memory ever used. |
| Memory Available | Memory currently available. |

The **show task memory** command does not display all the memory used by the routing protocol process. This value does not account for the memory used for the **TEXT** and

STACK segments, or the memory used by the routing protocol process's internal memory manager.

Why is the Currently In Use value less than the RES value?

The **show task memory** command displays a **Currently In Use** value measured in kilobytes. This value represents the memory currently in use. It is the dynamically allocated memory plus the **DATA** segment memory. The **show system processes extensive** command displays a **RES** value measured in kilobytes. This value represents the amount of program memory resident in the physical memory. This is also known as RSS or Resident Set Size.

The **Currently In Use** value does not account for all of the memory that the routing protocol process uses. This value does not include the memory used for the **TEXT** and the **STACK** segments, and a small percentage of memory used by the routing protocol process's internal memory manager. Further, the **RES** value includes shared library pages used by the routing protocol process.

Any amount of memory freed by the routing protocol process might still be considered part of the **RES** value. Generally, the kernel delays the migrating of memory out of the **Inact** queue into the **Cache** or **Free** list unless there is a memory shortage. This can lead to large discrepancies between the **Currently In Use** value and the **RES** value.

Frequently Asked Questions: Routing Protocol Process Memory Swapping

This section presents frequently asked questions and answers related to the memory swapping of the routing protocol process from the Routing Engine memory to the hard disk memory.

How do I monitor swap activity?

When the system is under memory pressure, the pageout process reuses memory from the free, cache, inactive and, if necessary, active pages. You can monitor the swap activity by viewing the syslog message reported by the kernel during periods of high pageout activity.

The syslog message appears as follows:

```
Mar  3 20:08:02 olympic /kernel: High pageout rate!! 277 pages/sec.
```

You can use the **vmstat -s** command to print the statistics for the swapout activity. The displayed statistics appear as follows:

```
0 swap pager pageouts
0 swap pager pages paged out
```

The **swap pager pageouts** is the number of pageout operations to the swap device, and the **swap pager pages paged out** is the number of pages paged out to the swap device.

Why does the system start swapping when I try to dump core using the request system core-dumps command?

The **request system core-dumps** command displays a list of system core files created when the device has failed. This command can be useful for diagnostic purposes. Each list item includes the file permissions, number of links, owner, group, size, modification

date, path, and filename. You can use the **core-filename** option and the **core-file-info**, **brief**, and **detail** options to display more information about the specified core-dump files.

You can use the **request system core-dumps** command to perform a non-fatal core-dump without aborting the routing protocol process. To do this, the routing protocol process is forked, generating a second copy, and then aborted. This process can double the memory consumed by the two copies of the routing protocol processes, pushing the system into swap.

Why does the show system processes extensive command show that memory is swapped to disk although there is plenty of free memory?

Memory can remain swapped out indefinitely if it is not accessed again. Therefore, the **show system processes extensive** command shows that memory is swapped to disk even though there is plenty of free memory, and such a situation is not unusual.

Frequently Asked Questions: Troubleshooting the Routing Protocol Process

This section presents frequently asked questions and answers related to a shortage of memory and memory leakage by the routing protocol process.

What does the RPD_OS_MEMHIGH message mean?

The **RPD_OS_MEMHIGH** message is written into the system message file if the routing protocol process is running out of memory. This message alerts you that the routing protocol process is using the indicated amount and percentage of Routing Engine memory, which is considered excessive. This message is generated either because the routing protocol process is leaking memory or the use of system resources is excessive, perhaps because routing filters are misconfigured or the configured network topology is very complex.

When the memory utilization for the routing protocol process is using all available Routing Engine DRAM memory (Routing Engines with maximum 2 GB DRAM) or reaches the limit of 2 GB of memory (Routing Engines with 4 GB DRAM), a message of the following form is written every minute in the syslog message file:

RPD_OS_MEMHIGH: Using 188830 KB of memory, 100 percent of available

This message includes the amount, in kilobytes and/or the percentage, of the available memory in use.

This message should not appear under normal conditions, as any further memory allocations usually require a portion of existing memory to be written to swap. As a recommended solution, increase the amount of RAM in the Routing Engine. For more information, go to <http://kb.juniper.net/InfoCenter/index?page=content&id=KB14186>.

What can I do when there is a memory shortage even after a swap?

It is not recommended for the system to operate in this state, notwithstanding the existence of swap. The protocols that run in the routing protocol process usually have a real-time requirement that cannot reliably withstand the latency of being swapped to hard disk. If the memory shortage has not resulted from a memory leak, then either a

reduction in the memory usage or an upgrade to a higher memory-capacity Routing Engine is required.

How do I determine whether there is a memory leak in the routing protocol process?

Memory leaks are typically the result of a seemingly unbounded growth in the memory usage of a process as reported by the **show system processes extensive** command.

There are two classes of memory leaks that the routing protocol process can experience.

- The first class occurs when the allocated memory that is no longer in use is not freed. This class of leak can usually be fixed by taking several samples of the **show task memory detail** command over a period of time and comparing the deltas.
- The second class occurs when there is a late access to freed memory. If the access is not outside the mapped address space, the kernel backfills the accessed page with real memory. This backfill is done without the knowledge of the routing protocol process's internal memory allocator, which makes this class of leak much more difficult to resolve. If a memory leak of this class is suspected, writing the state of the system to a disk file (creating a core file) is suggested.

A large discrepancy between the **RES** value and the **Currently In Use** value might indicate a memory leak. However, large discrepancies can also occur for legitimate reasons. For example, the memory used for the **TEXT** and **STACK** segments or the memory used by the routing protocol process's internal memory manager might not be displayed. Further, the **RES** value includes shared library pages used by the process.

What is the task_timer?

The source of a routing protocol process memory leak can usually be identified by dumping the timers for each task. You can use the **show task task-name** command to display routing protocol tasks on the Routing Engine. Tasks can be baseline tasks performed regardless of the device's configuration, and other tasks that depend on the device configuration.

For more information, see the **show task** command.

Related Documentation

- [Routing Protocol Process Memory FAQs Overview on page 53](#)

PART 5

Index

- [Index on page 63](#)

Index

Symbols

| | |
|--|---|
| #, comments in configuration statements..... | x |
| (), in syntax descriptions..... | x |
| < >, in syntax descriptions..... | x |
| [], in configuration statements..... | x |
| { }, in configuration statements..... | x |
| (pipe), in syntax descriptions..... | x |

B

| | |
|--|---|
| braces, in configuration statements..... | x |
| brackets | |
| angle, in syntax descriptions..... | x |
| square, in configuration statements..... | x |

C

| | |
|--|----|
| comments, in configuration statements..... | x |
| connections | |
| testing | |
| general connections..... | 34 |
| conventions | |
| text and syntax..... | ix |
| cryptographic-address statement..... | 11 |
| usage guidelines..... | 7 |
| curly braces, in configuration statements..... | x |
| customer support..... | xi |
| contacting JTAC..... | xi |

D

| | |
|------------------|----|
| documentation | |
| comments on..... | xi |

F

| | |
|--------------------------------------|--------|
| FAQs | |
| routing protocol process memory..... | 53, 54 |
| font conventions..... | ix |

H

| | |
|--------------------------|----|
| hosts, reachability | |
| general connections..... | 34 |

I

| | |
|--|----|
| interface statistics, real-time, displaying..... | 22 |
| IPv6 | |
| neighbor cache information | |
| displaying..... | 38 |
| router advertisements | |
| displaying..... | 40 |

K

| | |
|--|----|
| key-length statement..... | 12 |
| usage guidelines..... | 7 |
| key-pair statement..... | 12 |
| usage guidelines..... | 7 |
| keyboard sequences | |
| used with monitor interface command..... | 22 |
| used with monitor interface traffic | |
| command..... | 23 |

L

| | |
|---------------------------|----|
| log files | |
| contents, displaying..... | 43 |
| display of | |
| starting..... | 31 |
| stopping..... | 33 |

M

| | |
|--------------------------------|----|
| manuals | |
| comments on..... | xi |
| monitor interface command..... | 22 |
| monitor start command..... | 31 |
| monitor stop command..... | 33 |

N

| | |
|-----------------------------------|----|
| neighbor-discovery statement..... | 13 |
| usage guidelines..... | 7 |

O

| | |
|--|----|
| output control keys | |
| for monitor interface command..... | 22 |
| for monitor interface traffic command..... | 23 |

P

| | |
|--|----|
| parentheses, in syntax descriptions..... | x |
| ping command..... | 34 |

R

| | |
|----------------------|----|
| real-time monitoring | |
| interfaces..... | 22 |

| | |
|---------------------------------|--------|
| router advertisements | |
| IPv6 | |
| displaying..... | 40 |
| routes, displaying | |
| to specified network host..... | 46 |
| routing protocol process memory | |
| FAQ..... | 53, 54 |

S

| | |
|---|----|
| Secure Neighbor Discovery | |
| cryptographic addresses | |
| configuring..... | 7 |
| cryptographic-address statement..... | 11 |
| enabling..... | 7 |
| neighbor-discovery statement..... | 13 |
| security-level statement..... | 15 |
| timestamp statement..... | 16 |
| secure statement..... | 14 |
| security-level statement..... | 15 |
| show ipv6 neighbors command..... | 38 |
| show ipv6 router-advertisement command..... | 40 |
| show log command..... | 43 |
| statistics | |
| interfaces, real-time..... | 22 |
| support, technical See technical support | |
| syntax conventions..... | ix |

T

| | |
|--------------------------------|----|
| technical support | |
| contacting JTAC..... | xi |
| timestamp statement..... | 16 |
| trace files | |
| display of | |
| starting..... | 31 |
| stopping..... | 33 |
| traceoptions statement | |
| Secure Neighbor Discovery..... | 17 |
| traceroute command..... | 46 |

U

| | |
|-----------------------|----|
| users | |
| logs, displaying..... | 43 |