JUNIPER NETWORKS | Engineering Simplicity

Junos® OS

Flow-Based and Packet-Based Processing User Guide for Security Devices

Published
2020-03-25

JUNOS

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

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

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

Junos® OS Flow-Based and Packet-Based Processing User Guide for Security Devices
Copyright © 2020 Juniper Networks, Inc. All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

About the Documentation | xvii

Documentation and Release Notes | **xvii**

Using the Examples in This Manual | **xvii**

- Merging a Full Example | **xviii**
- Merging a Snippet | **xix**

Documentation Conventions | **xix**

Documentation Feedback | **xxii**

Requesting Technical Support | **xxii**

- Self-Help Online Tools and Resources | **xxiii**
- Creating a Service Request with JTAC | **xxiii**

1 **Overview**

Traffic Processing on SRX Series Devices Overview | 25

Understanding Traffic Processing on Security Devices | **25**

- Understanding Flow-Based Processing | **26**
- Understanding Packet-Based Processing | **28**

Understanding the Default Processing Behavior for IPv4 Traffic | **29**

Understanding Traffic Processing on SRX210 and SRX320 Devices | **30**

- Understanding Flow Processing and Session Management | **30**
- Understanding First-Packet Processing | **31**
- Understanding Session Creation | **31**
- Understanding Fast-Path Processing | **31**

Understanding Traffic Processing on SRX3000 Line and SRX1400 Devices | **32**

- Components Involved in Setting Up a Session | **33**
- Understanding the Data Path for Unicast Sessions | **34**
- Session Lookup and Packet Match Criteria | **34**
- Understanding Session Creation: First Packet Processing | **34**

Understanding Fast-Path Processing | 36

Understanding Traffic Processing on SRX4600 Devices | 36

Understanding Deployment Scenarios for the SRX4600 Services Gateway and Its Features | 37

Flow-Based Processing and Session Fundamentals | 39

Flow and Session Underlying Components Implemented Across SRX Series Services Gateways | 40

Understanding Traffic Processing on SRX5000 Line Devices | 41

Understanding First-Packet Processing | 42

Understanding Fast-Path Processing | 43

Understanding the Data Path for Unicast Sessions | 44

Understanding Services Processing Units | 50

Understanding Scheduler Characteristics | 51

Understanding Network Processor Bundling | 51

Configuring IOC to NPC Mapping | 52

Understanding Flow Processing on SRX5K-SPC3 Devices | 53

Understanding SPC3 Software Architecture | 54

Understanding Load Distribution | 55

Understanding NP Session and Service Offload (SOF) | 57

Understanding J-Flow support on SPC3 | 57

Understanding Datapath Debug SPU Support (E2E) | 57

Understanding Fragmentation Handling, ISSU, and ISHU Support | 58

Central Point Architecture in Security Devices Overview | 59

Understanding SRX Series Services Gateways Central Point Architecture | 59

Load Distribution in mixed Mode | 61

Sharing Processing Power and Memory in mixed Mode | 61

Understanding Enhancements to Central Point Architecture for the SRX5000 Line | 62

Understanding Central Point Session Limit Performance Enhancements | 63

Understanding Central Point Architecture Flow Support for GTP and SCTP | 64

Understanding the Flow Session Connection Filter Option | 65

2

Flow-Based Sessions

Flow-Based Sessions | 68

Understanding Session Characteristics for SRX Series Services Gateways | **68**

Understanding Aggressive Session Aging | **69**

Example: Controlling Session Termination for SRX Series Services Gateways | **69**

Clearing Sessions for SRX Series Services Gateways | **72**

Terminating Sessions for SRX Series Services Gateways | **72**

Terminating a Specific Session for SRX Series Services Gateways | **72**

Using Filters to Specify the Sessions to Be Terminated for SRX Series Services Gateways | **72**

Configuring the Timeout Value for Multicast Flow Sessions | **73**

TCP Sessions | 75

Understanding TCP Session Checks per Policy | **75**

Disabling TCP Packet Security Checks | **76**

Example: Configuring TCP Packet Security Checks Per Policy | **76**

Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways | **78**

Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways | **79**

TCP Out-of-State Packet Drop Logging Overview | **81**

Understanding TCP Out-of-State Packet Drop Logging | **83**

Supported TCP Out-of-State Logging Features | **83**

Understanding How Preserving Incoming Fragmentation Characteristics Can Improve Throughput | **84**

ECMP Flow-Based Forwarding | 86

Understanding ECMP Flow-Based Forwarding | **86**

ECMP Implementation for Junos OS SRX Series Devices and vSRX Instances | **88**

ECMP for Reverse Traffic | **88**

Example: Configuring ECMP Flow-Based Forwarding | **89**

Flow-Based Performance | 96

Expanding Session Capacity by Device | **96**

Expanding Session Capacity on an SRX3400 or SRX3600 Device | **97**

Reverting to Default Session Capacity on an SRX5800 Device | **97**

Verifying the Current Session Capacity | **97**

Flow Distribution and Packet-Ordering | 99

Understanding Load Distribution in SRX5000 Line Devices | **99**

Calculating SPU ID | **100**

Hash-Based Forwarding on the SRX5K-MPC, SRX5K-MPC3-40G10G (IOC3), and the SRX5K-MPC3-100G10G (IOC3) | **100**

Understanding Packet-Ordering Function on SRX5000 Line Devices | **102**

Changing Packet-Ordering Mode on SRX5000 Line Devices | **103**

Understanding Session Distribution on SRX5000 Line Devices in Adaptive Mode | **104**

Fragmentation Packets with PowerMode IPsec | 107

Understanding PMI First Path and Fast Path Processing | **107**

Switching between PMI First Path and Fast Path Processing | **107**

Fragmentation for Incoming IP Packets | **108**

Fragmentation for Outgoing IP Packets | **108**

NP session support | **108**

Unified Policies Support for Flow | 109

Flow First Path for Unified Policies | **110**

Understanding Flow Fast Path | **111**

Configuring the Session Log for the Default Security Policy | **112**

Configuring the Session Timeout for the Default Security Policy | **112**

TAP Mode for Flow Sessions | 113

Understanding TAP Mode Support for Security Flow Sessions | **114**

Example: Configuring Security Flow Sessions in TAP mode | **114**

Flow Management in SRX Series Devices Using VRF Routing Instance | 117

Virtual Routing and Forwarding Instances in SD-WAN Deployments | **117**

Flow Management Using VRF Routing Instance | **118**

Virtual Routing and Forwarding Groups | **119**

Understanding VRF groups | **120**

Types of VRF groups | **121**

VRF Movement | **121**

VRF group-ID | **122**

Configuring VRF groups | **122**

VRF group Operations | **123**

Flow Processing using Virtual Routing and Forwarding Group | **124**

First Path Processing using VRF Group | **124**

Fast Path Processing using VRF Group | **126**

Example: Configuring a Security Policy to Permit VRF-Based Traffic from an IP Network to MPLS Network using VRF Group | **127**

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to an IP Network using VRF Group | **131**

Example: Configuring a Security Policy to Permit VRF-Based Traffic from Public IP Network to MPLS Network using VRF Group | **135**

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to Public IP Network to using VRF Group | **142**

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to MPLS Network without NAT using VRF Group | **149**

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to MPLS Network using NAT and VRF Group | **154**

3 **Flow-Based Processing for IPv6**

IPv6 Flow-Based Processing | 163

IPv6 Advanced Flow | **163**

Understanding Sessions for IPv6 Flow | **165**

Understanding IPv6 Flow Processing on SRX5400, SRX5600, and SRX5800 devices | **165**

Enabling Flow-Based Processing for IPv6 Traffic | **168**

Flow-Based Processing for IPv6 Traffic on Security Devices | **169**

Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways | **171**

Understanding Path MTU Messages for IPv6 Packets | 178

IPv6 Packets Header Overview | 180

The IPv6 Packet Header and SRX Series Overview | **180**

Understanding IPv6 Packet Header Extensions | **180**

Understanding How SRX Series Devices Handle ICMPv6 Packets | **182**

4 Monitoring Flow-Based Sessions and Establishing Parameters for Error Handling

Monitoring Security Flow Sessions | 186

Monitoring Security Flow Sessions Overview | 186

Understanding How to Obtain Session Information for SRX Series Services Gateways | 187

Displaying Global Session Parameters for All SRX Series Services Gateways | 189

Displaying a Summary of Sessions for SRX Series Services Gateways | 189

Displaying Session and Flow Information About Sessions for SRX Series Services Gateways | 190

Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways | 190

Using Filters to Display Session and Flow Information for SRX Series Services Gateways | 191

Information Provided in Session Log Entries for SRX Series Services Gateways | 191

Error Handling Extensions | 196

Understanding Chassis Manager FPC Fault Detection and Error Handling Enhancements | 197

Monitoring X2 Traffic | 200

Understanding X2 Traffic Monitoring | 201

X2 Traffic Monitoring Overview | 201

Limitations of X2 Traffic Monitoring | 203

X2 Traffic Terminology | 203

Example: Configuring a Mirror Filter for X2 Traffic Monitoring | 204

5 Packet Based Forwarding

Packet-Based Forwarding | 211

Understanding Packet-Based Processing | 211

Understanding Selective Stateless Packet-Based Services | 212

Selective Stateless Packet-Based Services Configuration Overview | 214

Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding | 216

Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding | 228

Understanding Session Cache | 241

Overview | 241

Selective Session Cache Installation | 242

IPsec VPN Session Affinity Enhancement Using Session Cache | 243

Fragmentation Packet Ordering Using NP Session Cache | **244**

Understanding Symmetric Fat IPsec Tunnel | **245**

Reverse Route Packet Mode using Virtual Router | 246

Understanding To-host Traffic on Virtual Router | **247**

Express Path | 249

Express Path Overview | **249**

Understanding Express Path Functionality | **250**

Understanding Express Path Support on SRX Series Devices | **251**

Understanding Express Path Features | **252**

Express Path Limitations | **257**

Express Path Support on NP-IOC Card | **259**

Express Path Support on SRX5K Modular Port Concentrator | **259**

Express Path Support on SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) | **260**

IPv6 Flow in Express Path Mode for IOC2 and IOC3 | **263**

IPv6 Flow in Express Path Mode | **264**

Understanding the Express Path Solution | **265**

Enabling and Disabling Express Path | **266**

Example: Enabling Express Path in Security Policies | **267**

Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path | **269**

Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path | **272**

Example: Configuring SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) on an SRX5000 Line Device to Support Express Path | **275**

Example: Configuring Express Path on an SRX5000 Line Device with IOC3 | **278**

Example: Configuring Low Latency | **284**

Managing Packet Fragmentation in IPsec VPN Networks | **287**

Fragmentation Counters Feature Overview | **288**

Understanding Fragmentation and MTU and MSS Sizes | **288**

Using Fragmentation Counter Statistics to Tune Your System | **289**

6 **Configuration Statements**

aging | 294

all-tcp | 295

allow-dns-reply | 296

allow-embedded-icmp | 297

allow-reverse-ecmp | 298

application-services (Security Forwarding Process) | 299

apply-to-half-close-state | 301

destination-header | 302

destination-port (Security Forwarding Options) | 304

destination-prefix (Security Forwarding Options) | 308

early-ageout | 309

error | 310

fin-invalidate-session | 314

flow (Security Flow) | 316

force-ip-reassembly | 322

forwarding-process | 323

fpc error | 325

fru-poweron-sequence | 327

gre-in | 328

gre-out | 329

high-watermark | 330

hop-by-hop-header | 331

icmpv6-malformed | 332

idle-timeout (System Services) | 333

inline-tap | 334

interface-in (Security Forwarding Options) | 335

interface-out (Security Forwarding Options) | 336

ipv4-template (Services) | 337

ipv6-extension-header | 338

ipv6-extension-header-limit | 340

ipv6-malformed-header | 341

ipv6-template (Services) | 342

low-latency | 343

low-watermark | 344

maximize-idp-sessions | 345

mirror-filter (Security Forwarding Options) | 346

mode (Security Forwarding Options) | 348

no-sequence-check | 350

np-cache (Flexible PIC Concentrator) | 351

output (Security Forwarding Options) | 352

packet-filter | 353

packet-log (Security Flow) | 355

packet-ordering-mode (Application Services) | 356

pending-sess-queue-length | 357

per-session-statistics | 358

pre-id-default-policy | 359

preserve-incoming-fragment-size | 361

propagate-settings | 362

protocol (Security Forwarding Options) | 363

resource-manager | 365

reverse-route-packet-mode-vr | 366

route-change-timeout | 367

rst-invalidate-session | 368

rst-sequence-check | 369

sampling | 370

services-offload | 371

session (System Services) | 372

session-limit (System Services) | 373

source-port (Security Forwarding Options) | 374

source-prefix (Security Forwarding Options) | 378

syn-flood-protection-mode | 379

tcp-initial-timeout | 380

tcp-mss (Security Flow) | 381

tcp-session | 383

time-wait-state | 385

traceoptions (Security) | 386

traceoptions (Security Flow) | 388

transport (Security Log) | 391

weight (Security) | 393

7 Operational Commands

clear firewall | 399

clear monitor security flow filter | 401

clear security flow ip-action | 402

clear security flow session all | 405

clear security flow session application | 407

clear security flow session application-traffic-control | 410

clear security flow session conn-tag | 413

clear security flow session destination-port | 415

clear security flow session destination-prefix | 417

clear security flow session family | 419

clear security flow session IDP | 421

clear security flow session interface | 424

clear security flow ip-action | 426

clear security flow session nat | 429

clear security flow session protocol | 432

clear security flow session resource-manager | 435

clear security flow session services-offload | 437

clear security flow session session-identifier | 441

clear security flow session source-port | 443

clear security flow session source-prefix | 445

clear security flow session tunnel | 447

clear security forward-options mirror filter | 450

monitor security flow file | 451

monitor security flow filter | 453

monitor security flow start | 455

monitor security flow stop | 456

show chassis environment (Security) | 457

show chassis fpc (View) | 464

show chassis fpc errors | 475

show chassis hardware (View) | 478

show chassis pic (Security) | 499

show chassis power | 502

show chassis power sequence | 507

show firewall (View) | 509

show interfaces (View Aggregated Ethernet) | 512

show interfaces diagnostics optics | 526

show interfaces flow-statistics | 532

show interfaces swfabx | 538

show monitor security flow | 540

show security flow cp-session | 542

show security flow cp-session destination-port | 546

show security flow cp-session destination-prefix | 549

show security flow cp-session family | 552

show security flow cp-session protocol | 555

show security flow cp-session source-port | 558

show security flow cp-session source-prefix | 561

show security flow gate | 564

show security flow ip-action | 569

show security flow gate brief node | 580

show security flow gate destination-port | 588

show security flow gate destination-prefix | 592

show security flow gate protocol | 596

show security flow gate summary node | 600

show security flow pmi statistics | 607

show security flow session | 610

show security flow session brief node | 620

show security flow session destination-port | 625

show security flow session destination-prefix | 630

show security flow session extensive node | 636

show security flow session family | 644

show security flow session interface | 651

show security flow session nat | 656

show security flow session policy-id | 660

show security flow session protocol | 664

show security flow session resource-manager | 670

show security flow session services-offload | 675

show security flow session session-identifier | 682

show security flow session source-port | 688

show security flow session source-prefix | 693

show security flow session summary family | 698

show security flow session summary node | 701

show security flow session summary services-offload | 709

show security flow session tunnel | 714

show security flow statistics | 725

show security flow status | 731

show security forwarding-options mirror-filter | 736

show security monitoring | 739

show security policies | 742

show security policies hit-count | 759

show security resource-manager group active | 763

show security resource-manager resource active | 767

show security resource-manager settings | 771

show security resource-manager summary | 774

show security screen ids-option | 776

show security screen statistics | 783

show security softwires | 797

show security zones | 799

show security zones type | 807

About the Documentation

IN THIS SECTION

- Documentation and Release Notes | **xvii**
- Using the Examples in This Manual | **xvii**
- Documentation Conventions | **xix**
- Documentation Feedback | **xxii**
- Requesting Technical Support | **xxii**

Use this guide to configure and monitor the flow of traffic or packet, on a device using flow-based processing and packet-based forwarding. Also, for using an extensive set of flow-based security features which include policies, screens, network address translation (NAT), and other flow-based services.

Documentation and Release Notes

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

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

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

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

   ```
   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.xsl; }
   ```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

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

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

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

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

Documentation Conventions

Table 1 on page xx 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.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page xx 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`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*

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

Convention	Description	Examples
Italic text like this	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** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• 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)	Encloses 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** **(*string1* | *string2* | *string3*)**
# (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)	Encloses a variable for which you can substitute one or more values.	**community name members [*community-ids*]**

Indention and braces ({ })	Identifies a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		

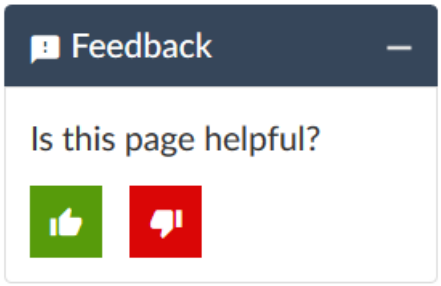
Table 2: Text and Syntax Conventions *(continued)*

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

Documentation Feedback

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

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the Juniper Networks TechLibrary site, and do one of the following:



  - Click the thumbs-up icon if the information on the page was helpful to you.
  - Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.
- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services support contract, or are

covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit https://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

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

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

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

Creating a Service Request with JTAC

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

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

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

1

CHAPTER

Overview

Traffic Processing on SRX Series Devices Overview | 25

Central Point Architecture in Security Devices Overview | 59

Traffic Processing on SRX Series Devices Overview

IN THIS SECTION

- Understanding Traffic Processing on Security Devices | 25
- Understanding the Default Processing Behavior for IPv4 Traffic | 29
- Understanding Traffic Processing on SRX210 and SRX320 Devices | 30
- Understanding Traffic Processing on SRX3000 Line and SRX1400 Devices | 32
- Understanding Traffic Processing on SRX4600 Devices | 36
- Understanding Traffic Processing on SRX5000 Line Devices | 41
- Configuring IOC to NPC Mapping | 52
- Understanding Flow Processing on SRX5K-SPC3 Devices | 53

Junos OS for security devices integrates network security and routing capabilities of Juniper Networks. Packets that enter and exit a device undergo both packet-based and flow-based processing.

Understanding Traffic Processing on Security Devices

IN THIS SECTION

- Understanding Flow-Based Processing | 26
- Understanding Packet-Based Processing | 28

Junos OS for security devices integrates the world-class network security and routing capabilities of Juniper Networks. Junos OS includes a wide range of packet-based filtering, class-of-service (CoS) classifiers, and traffic-shaping features as well as a rich, extensive set of flow-based security features including policies, screens, network address translation (NAT), and other flow-based services.

Traffic that enters and exits a security device is processed according to features you configure, such as packet filters, security policies, and screens. For example, the software can determine:

- Whether the packet is allowed into the device
- Which firewall screens to apply to the packet
- The route the packet takes to reach its destination
- Which CoS to apply to the packet, if any
- Whether to apply NAT to translate the packet's IP address
- Whether the packet requires an Application Layer Gateway (ALG)

Packets that enter and exit a device undergo both packet-based and flow-based processing:

- Flow-based packet processing treats related packets, or a stream of packets, in the same way. Packet treatment depends on characteristics that were established for the first packet of the packet stream, which is referred to as a flow.

  For the distributed processing architecture of the services gateway, all flow-based processing occurs on the SPU and sampling is multi-thread aware. Packet sequencing is maintained for the sampled packets.

- Packet-based, or stateless, packet processing treats packets discretely. Each packet is assessed individually for treatment.

  For the distributed processing architecture of the services gateway, some packet-based processing, such as traffic shaping, occurs on the NPU. Some packet-based processing, such as application of classifiers to a packet, occurs on the SPU.

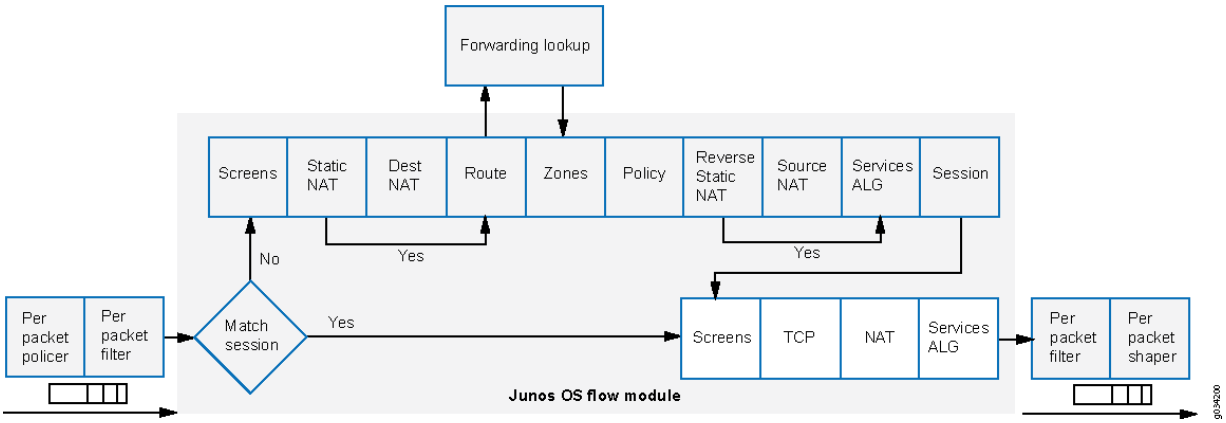
This topic includes the following sections:

Understanding Flow-Based Processing

A packet undergoes flow-based processing after packet-based filters and some screens have been applied to it. All flow-based processing for a single flow occurs on a single Services Processing Unit (SPU). An SPU processes the packets of a flow according to the security features and other services configured for the session.

Figure 1 on page 27 shows a conceptual view of how flow-based traffic processing occurs on services gateway.

Figure 1: Traffic Flow for Flow-Based Processing



A flow is a stream of related packets that meet the same matching criteria and share the same characteristics. Junos OS treats packets belonging to the same flow in the same manner.

Configuration settings that determine the fate of a packet—such as the security policy that applies to it, if it requires an Application Layer Gateway (ALG), if NAT is applied to translate the packet's source and/or destination IP address—are assessed for the first packet of a flow.

To determine if a flow exists for a packet, the NPU attempts to match the packet's information to that of an existing session based on the following match criteria:

- Source address
- Destination address
- Source port
- Destination port
- Protocol
- Unique session token number for a given zone and virtual router

Zones and Policies

The security policy to be used for the first packet of a flow is cached in a flow table for use with the same flow and closely related flows. Security policies are associated with zones. A zone is a collection of interfaces that define a security boundary. A packet's incoming zone, as determined by the interface through which it arrived, and its outgoing zone, as determined by the forwarding lookup, together determine which policy is used for packets of the flow.

Flows and Sessions

Flow-based packet processing, which is stateful, requires the creation of sessions. A session is created for the first packet of a flow for the following purposes:

- To store most of the security measures to be applied to the packets of the flow.
- To cache information about the state of the flow.

  For example, logging and counting information for a flow is cached in its session. (Some stateful firewall screens rely on threshold values that pertain to individual sessions or across all sessions.)
- To allocate required resources for the flow for features such as NAT.
- To provide a framework for features such as ALGs and firewall features.

Most packet processing occurs in the context of a flow, including:

- Management of policies, NAT, zones, and most screens.
- Management of ALGs and authentication.

Understanding Packet-Based Processing

A packet undergoes packet-based processing when it is removed from the queue on its input interface and before it is added to the queue on its output interface.

Packet-based processing applies stateless firewall filters, CoS features, and some screens to discrete packets.

- When a packet arrives at an interface, sanity checks, packet-based filters, some CoS features, and some screens are applied to it.
- Before a packet leaves the device, any packet-based filters, some CoS features, and some screens associated with the interface are applied to the packet.

Filters and CoS features are typically associated with one or more interfaces to influence which packets are allowed to transit the system and to apply special actions to packets as necessary.

The following topics describe the kinds of packet-based features that you can configure and apply to transit traffic.

Stateless Firewall Filters

Also referred to as access control lists (ACLs), stateless firewall filters control access and limit traffic rates. They statically evaluate the contents of packets transiting the device from a source to a destination, or packets originating from or destined for the Routing Engine. A stateless firewall filter evaluates every packet, including fragmented packets.

You can apply a stateless firewall filter to an input or output interface, or to both. A filter contains one or more terms, and each term consists of two components—match conditions and actions. By default, a packet that does not match a firewall filter is discarded.

You can plan and design stateless firewall filters to be used for various purposes—for example, to limit traffic to certain protocols, IP source or destination addresses, or data rates. Stateless firewall filters are executed on the NPU.

Class-of-Service Features

CoS features allow you to classify and shape traffic. CoS features are executed on the NPU.

- Behavior aggregate (BA) classifiers—These classifiers operate on packets as they enter the device. Using behavior aggregate classifiers, the device aggregates different types of traffic into a single forwarding class to receive the same forwarding treatment. BA classifiers allow you to set the forwarding class and loss priority of a packet based on the Differentiated Service (DiffServ) value.
- Traffic shaping—You can shape traffic by assigning service levels with different delay, jitter, and packet loss characteristics to particular applications served by specific traffic flows. Traffic shaping is especially useful for real-time applications, such as voice and video transmission.

Screens

Some screens, such as denial-of-service (DoS) screens, are applied to a packet outside the flow process. They are executed on the Network Processing Unit (NPU).

Understanding the Default Processing Behavior for IPv4 Traffic

Flow-based processing mode is required for security features such as zones, screens, and firewall policies to function. By default, the SRX Series device is enabled for flow-based forwarding for IPv4 traffic on all devices, apart from the SRX300 Series and SRX550M devices that are set to drop mode. Starting with Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, for the SRX1500 series, SRX4100, SRX4200, SRX5400, SRX5600, SRX5800 and vSRX devices, you do *not* need to reboot the device when you are switching modes between flow mode, packet mode, and drop mode. For SRX300 Series and SRX550M devices, you *must* reboot the device when switching between flow mode, packet mode, and drop mode.

SRX300 Series and SRX550M

For the SRX300 Series and the SRX550M devices, the default processing mode is set to drop mode because of memory constraints. In this case, you must reboot the device after changing the processing mode from the drop mode default to flow-based processing mode or packet-based processing mode—that is, between modes on these devices.

> **NOTE:** For drop mode processing, the traffic is dropped directly, it is not forwarded. It differs from packet-mode processing for which the traffic is handled but no security processes are applied.

Configuring an SRX Series Device as a Border Router

When an SRX Series device of any type is enabled for flow-based processing or drop mode, to configure the device as a border router you must change the mode to packet-based processing for MPLS. In this case, to configure the SRX device to packet mode for MPLS, use the **set security forwarding-options family mpls mode packet-based** statement.

> **NOTE:** As mentioned previously, for SRX300 Series and the SRX550M devices, whenever you change processing modes, you must reboot the device.

Understanding Traffic Processing on SRX210 and SRX320 Devices

IN THIS SECTION

- Understanding Flow Processing and Session Management | 30
- Understanding First-Packet Processing | 31
- Understanding Session Creation | 31
- Understanding Fast-Path Processing | 31

This topic describes the process that the SRX210 and SRX320 Services Gateways undertake in establishing a session for packets belonging to a flow that transits the device. The flow services of the SRX210 and SRX320 devices are single-threaded and non-distributed. Although they differ from the other SRX Series devices in this respect, the same flow model is followed and the same command line interface (CLI) is implemented.

To illustrate session establishment and the packet "walk" including the points at which services are applied to the packets of a flow, the example described in the following sections uses the simple case of a unicast session:

Understanding Flow Processing and Session Management

This topic explains how a session is set up to process the packets composing a flow. In the following topic, the SPU refers to the data plane thread of the SRX210 or SRX320 Services Gateway.

At the outset, the data plane thread fetches the packet and performs basic sanity checks on it. Then it processes the packet for stateless filters and CoS classifiers and applies some screens.

Understanding First-Packet Processing

To determine if a packet belongs to an existing flow, the device attempts to match the packet's information to that of an existing session based on the following six match criteria:

- Source address
- Destination address
- Source port
- Destination port
- Protocol
- Unique token from a given zone and virtual router

The SPU checks its session table for an existing session for the packet. If no existent session is found, the SPU sets up a session for the flow. If a session match is found, the session has already been created, so the SPU performs fast-path processing on the packet.

Understanding Session Creation

In setting up the session, the SPU executes the following services for the packet:

- Screens
- Route lookup
- Policy lookup
- Service lookup
- NAT, if required

After a session is set up, it is used for all packets belonging to the flow. Packets of a flow are processed according to the parameters of its session. For the remainder of the steps entailed in packet processing, proceed to Step 1 in "Fast-Path Processing". All packets undergo fast-path processing.

Understanding Fast-Path Processing

If a packet matches a session, Junos OS performs fast-path processing as described in the following steps. After a session has been set up for the first packet in a flow, also undergoes fast-path processing. All packets undergo fast-path processing.

1. The SPU applies flow-based security features to the packet.

- Configured screens are applied.
- TCP checks are performed.
- Flow services, such as NAT, ALG, and IPsec are applied, if required.

2. The SPU prepares the packet for forwarding and transmits it.
   - Routing packet filters are applied.
   - Traffic shaping is applied.
   - Traffic prioritizing is applied.
   - Traffic scheduling is applied.
   - The packet is transmitted.

Understanding Traffic Processing on SRX3000 Line and SRX1400 Devices

IN THIS SECTION

- Components Involved in Setting Up a Session | 33
- Understanding the Data Path for Unicast Sessions | 34
- Session Lookup and Packet Match Criteria | 34
- Understanding Session Creation: First Packet Processing | 34
- Understanding Fast-Path Processing | 36

Junos OS for the SRX1400, SRX3400 and SRX3600 Services Gateways integrates the world-class network security and routing capabilities of Juniper networks. Junos OS for these service gateways includes the wide range of security services including policies, screens, network address translation, class-of-service classifiers, and the rich, extensive set of flow-based services that are also supported on the other devices in the services gateways.

The distributed parallel processing architecture of the SRX1400, SRX3400 and SRX3600 devices includes multiple processors to manage sessions and run security and other services processing. This architecture provides greater flexibility and allows for high throughput and fast performance.

The following sections describe the processing architecture using SRX3400 and SRX3600 devices as an example:

This topic includes the following information:

Components Involved in Setting Up a Session

Here is an overview of the main components involved in setting up a session for a packet and processing the packets as they transit the SRX3400 and SRX3600 devices:

- Services Processing Units (SPUs)—The main processors of the SRX3400 and SRX3600 devices reside on Services Processing Cards (SPCs). They establish and manage traffic flows and perform most of the packet processing on a packet as it transits the device. Each SPU maintains a hash table for fast session lookup. The SPU performs all flow-based processing for a packet, including application of security services, classifiers, and traffic shapers. All packets that belong to the same flow are processed by the same SPU.

  The SPU maintains a session table with entries for all sessions that it established and whose packets it processes. When an SPU receives a packet from an NPU, it checks its session table to ensure that the packet belongs to it.

  For SRX3400 and SRX3600 devices, one SPU acts in concert performing its regular session management and flow processing functions and acting as a central point in which it arbitrates sessions and allocates resources. When an SPU performs in this manner it is said to be in combo mode.

- Central Point—The central point is used to allocate session management to SPUs based on load balancing criteria. It distributes sessions in an intelligent way to avoid occurrences in which multiple SPUs might wrongly handle the same flow. The central point follows load balancing criteria in allocating sessions to SPUs. If the session exists, the central point forwards packets for that flow to the SPU hosting it. It also redirects packets to the correct SPU in the event that the NPU fails to do so.

  For the SRX3400 and SRX3600 devices, one SPU always runs in what is referred to as combo mode in which it implements both the functionality of the central point and the flow and session management functionality. In combo mode, the SPU and the central point share the same load-balancing thread (LBT) and packet-ordering thread (POT) infrastructure. .

- Routing Engine (RE)—The Routing Engine runs the control plane and manages the Control Plane Processor (CPP).

Understanding the Data Path for Unicast Sessions

Junos OS for the SRX3400 and SRX3600 Services Gateways is a distributed parallel processing high throughput and high performance system. This topic describes the process of establishing a session for packets belonging to a flow that transits the device.

To illustrate session establishment and the packet "walk" including the points at which services are applied to the packets of a flow, the following example uses the simple case of a unicast session. This packet "walk" brings together the packet-based processing and flow-based processing that the Junos OS performs on the packet.

Session Lookup and Packet Match Criteria

To determine if a packet belongs to an existing flow, the device attempts to match the packet's information to that of an existing session based on the following six match criteria:

- Source address
- Destination address
- Source port
- Destination port
- Protocol
- Unique token from a given zone and virtual router

Understanding Session Creation: First Packet Processing

This topic explains how a session is set up to process the packets composing a flow. To illustrate the process, this topic uses an example with a source "a" and a destination "b". The direction from source to destination for the packets of the flow is referred to as (a -> b). The direction from destination to source is referred to as (b -> a).

1. A packet arrives at an interface on the device and the IOC processes it.

   The IOC dequeues the packet and sends it to the NPU with which it communicates.

2. The NPU receives the packet from the IOC and processes it.
   - The NPU performs basic sanity checks on the packet and applies some screens configured for the interface to the packet.

- If a session match is found, the session has already been created on an SPU that was assigned to it, so the NPU forwards the packet to the SPU for processing along with the session ID.

Example: Packet (a ->b) arrives at NPU1 from IOC1. NPU1 performs sanity checks and applies DoS screens to the packet. NPU1 checks its session table for a tuple match and no existing session is found. NPU1 forwards the packet to the central point on SPU1 for assignment to an SPU.

3. The central point creates a session with a “Pending” state.

   The central point maintains a global session table that includes entries for all sessions that exist across all SPUs on the device. It participates in session creation and delegates and arbitrates session resources allocation.

   This process entails the following parts:

   a. The central point checks its session table and gate table to determine if a session or a gate exists for the packet it receives from the NPU. (An NPU has forwarded a packet to the central point because its table indicates there is no session for it. The central point verifies this information before allocating an SPU for the session.)

   b. If there is no entry that matches the packet in either table, the central point creates a pending wing for the session and selects an SPU to be used for the session, based on its load-balancing algorithm.

   c. The central point forwards the first packet of the flow to the selected SPU in a message telling it to set up a session locally to be used for the packet flow.

      Example: The central point creates pending wing (a ->b) for the session. It selects SPU1 to be used for the session. It sends SPU1 the (a->b) packet along with a message to create a session for it. (It happens to be the case that SPU1 is the SPU that runs in combo mode. Therefore, its session-management and flow-processing services are used for the session.

4. The SPU sets up the session.

   Each SPU, too, has a session table, which contains information about its sessions. When the SPU receives a message from the central point to set up a session, it checks its session table to ensure that a session does not already exist for the packet.

   a. If there is no existing session for the packet, the SPU sets up the session locally.

   b. The SPU sends a message to the central point, telling it to install the session.

   During first-packet processing, if NAT is enabled, the SPU allocates IP address resources for NAT. In this case, the first-packet processing for the session is suspended until the NAT allocation process is completed.

   The SPU adds to the queue any additional packets for the flow that it might receive until the session has been installed.

   Example: SPU1 creates the session for (a ->b) and sends a message back to the central point (implemented on the same SPU) telling it to install the pending session.

5. The central point installs the session.

- It sets the state for the session's pending wing to active.
- It installs the reverse wing for the session as an active wing.
- It sends an ACK (acknowledge) message to the SPU, indicating that the session is installed.

Example: The central point receives a message from SPU1 to install the session for (a->b). It sets the session state for (a->b) wing to active. It installs the reverse wing (b->a) for the session and makes it active; this allows for delivery of packets from the reverse direction of the flow: destination (b) to be delivered to the source (a).

6. The SPU sets up the session on the ingress and egress NPUs.

   NPUs maintain information about a session for packet forwarding and delivery. Session information is set up on the egress and ingress NPUs (which sometimes are the same) so that packets can be sent directly to the SPU that manages their flows and not to the central point for redirection.

7. Fast-path processing takes place.

   For the remainder of the steps entailed in packet processing, proceed to Step 1 in "Understanding Fast-Path Processing".

Understanding Fast-Path Processing

All packets undergo fast-path processing. However, if a session exists for a packet, the packet undergoes fast-path processing and bypasses the first-packet process. When there is already a session for the packet's flow, the packet does not transit the central point.

Here is how fast-path processing works: NPUs at the egress and ingress interfaces contain session tables that include the identification of the SPU that manages a packet's flow. Because the NPUs have this session information, all traffic for the flow, including reverse traffic, is sent directly to that SPU for processing.

On SRX1400, SRX3400, and SRX3600 devices, the iflset functionality is not supported for aggregated interfaces like *reth*.

Understanding Traffic Processing on SRX4600 Devices

IN THIS SECTION

- Understanding Deployment Scenarios for the SRX4600 Services Gateway and Its Features | **37**
- Flow-Based Processing and Session Fundamentals | **39**
- Flow and Session Underlying Components Implemented Across SRX Series Services Gateways | **40**

The Juniper Networks SRX4600 Services Gateway integrates flow-based security and routing services, including advanced security and threat mitigation and traditional stateful firewall security. The Junos OS flow-based infrastructure provides the foundation and framework for Layer 4 through Layer 7 application-based services. The SRX4600 Services Gateway is designed to be deployed as an integrated firewall at the large enterprise data center edge and data center core, and the campus edge. It can also be deployed as an LTE security gateway and a Gi/SGi firewall.

This topic includes the following content:

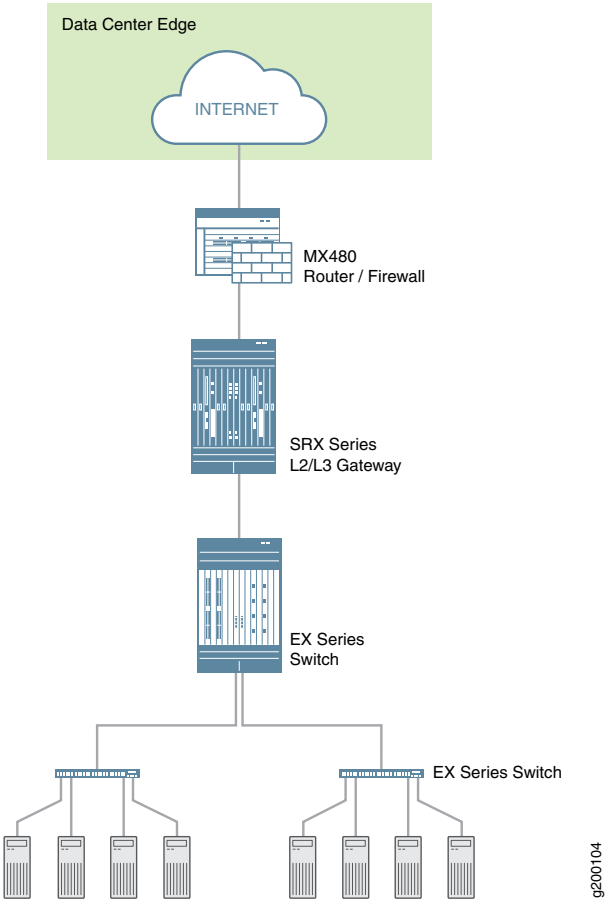
Understanding Deployment Scenarios for the SRX4600 Services Gateway and Its Features

The SRX4600 Services Gateway can be deployed in many areas to secure your environment and its resources. It is often used to protect the data center edge and core in the following ways:

- Deploying the SRX4600 Services Gateway as a Data Center Edge Firewall

  You can deploy the SRX4600 Services Gateway at the edge of your data center to provide the applications and services that it hosts with optimum protection. Every data center has an ingress point to allow clients access to the data center's services, but malicious aggressors can take advantage of it to launch attacks against these services. A large amount of traffic coming into the data center is ingress internet traffic. For that reason alone, deploying robust, multi-layered security at the data center edge is essential. The SRX4600 Services Gateway effectively and reliantly blocks attacks, and it allows you to configure the system to thwart specific kinds of attacks. The SRX4600 Services Gateway supports Juniper's Software-Defined Secure Network (SDSN) framework, including Sky Advanced Threat Prevention (Sky ATP), which is built around automated and actionable intelligence that can be shared quickly to recognize and mitigate threats. Figure 2 on page 38 shows the SRX4600 Services Gateway deployed at the data center edge in conjunction with an MX480 router and EX Series switches.

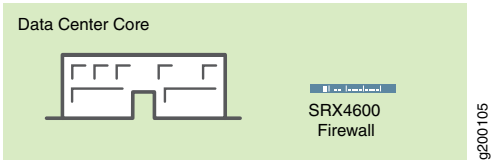
Figure 2: Deploying the SRX4600 Services Gateway at the Data Center Edge



- Deploying the SRX4600 Services Gateway at the Data Center Core

  You can deploy the SRX4600 Services Gateway at the data center core to provide enhanced security and to ensure that compliance requirements are met. Data center processing has become increasingly dynamic necessitating clear network definition and compliance requirements enforcement. To ensure compliance, you can use the SRX4600 Services Gateway to segment your overall network into individual server networks and secure traffic within them. The SRX4600 Services Gateway provides high availability and automation, and its high performance Layer 3 and Layer 4 services meet the security requirements of the data center core. Figure 3 on page 39 shows the SRX4600 Services Gateway deployed as a multi-layered firewall at the data center core.

Figure 3: Deploying the SRX4600 Services Gateway at the Data Center Core



In addition to its advanced anti-malware features, the SRX4600 Services Gateway supports the following features:

- Stateful firewall
- Application security suite
- UTM (Sophos AV, Web filtering, antispam)
- IDP
- High availability (Chassis cluster)
  - Dual HA control ports (10G)
  - MACsec support for HA ports
- Ethernet interfaces through QSFP28 (100G/40G/4x10G modes), QSFP+ (40G/4x10G modes) and SFP+ (10G mode)
- IPsec VPN, including AutoVPN and Group VPNv2
- QoS and network services
- J-Web
- Routing policies with multicast

Flow-Based Processing and Session Fundamentals

To understand flow processing on the SRX4600 Services Gateway, it is important to understand the fundamentals of flow.

A flow is a stream of related packets that meet the same matching criteria and share the same characteristics. Junos OS treats packets that belong to the same flow in the same way. The architecture of an SRX Series services gateway and how it handles packet flows are tightly coupled. Consequently, in part, flow is implemented differently across the family of SRX Series services gateways because of their architectural differences.

Flow-based packet processing, which is stateful, requires the creation of sessions. Sessions are created based on routing and other traffic classification information to store information and allocate resources

for a flow. Sessions cache information about the state of the flow, and they store most of the security measures to be applied to packets of the flow. Because of the architectural differences across devices, sessions are also managed differently by different devices.

Regardless of these differences, conceptually the flow process is the same across all services gateways, and sessions serve the same purposes and have the same features.

Flow and Session Underlying Components Implemented Across SRX Series Services Gateways

SRX Series services gateways use the same infrastructure components to support flow and manage sessions, but not all devices implement all of them.

To understand flow, it is essential to understand the following components and how they are used:

- The Services Processing Unit (SPU)

  An SPU manages the session for a packet flow. It applies security features and other services to the packet. It also applies packet-based stateless firewall filters, classifiers, and traffic shapers to the packet.

- The central point (CP)

  The central point is an SPU that the system uses to allocate resources and distribute session management among SPUs. When the first packet of a flow is processed, the central point determines which SPU to use for that packet's session. The SRX4600 Services Gateway does not implement a central point.

- The Network Processing Unit (NPU) and the Network Processing session

  An NPU is a processor that runs on an I/O card (IOC) and processes packets discretely. When a flow is created, subsequent packets of the flow are matched to the session on the NPU. The NPU handles additional processing such as TCP sequence check, time-to-live (TTL) processing, and Layer 2 header translation. An NPU improves performance in that extra packet forwarding between a session-SPU and a hash-SPU is avoided. The SRX4600 Services Gateway implements an NPU.

The SRX4600 Services Gateway flow architecture has been improved to optimize use of the SRX4600 device's advanced multi-core Xeon™ Processors. The SRX4600 Services Gateway implements the use of a dedicated session thread to circumvent problems such as management of out-of-order packets in a flow. It utilizes the network processing session to ensure that packets are forwarded to the right, dedicated thread. Packets are distributed to different threads in accord with the hash-based session distribution model.

Understanding Traffic Processing on SRX5000 Line Devices

IN THIS SECTION

- Understanding First-Packet Processing | 42
- Understanding Fast-Path Processing | 43
- Understanding the Data Path for Unicast Sessions | 44
- Understanding Services Processing Units | 50
- Understanding Scheduler Characteristics | 51
- Understanding Network Processor Bundling | 51

Junos OS on SRX5000 devices is a distributed, parallel processing, high-throughput and high-performance system. The distributed parallel processing architecture of the SRX5000 line of services gateways includes multiple processors to manage sessions and run security and other services processing. This architecture provides greater flexibility and allows for high throughput and fast performance.

> **NOTE:** In SRX1400, SRX3400, SRX3600, SRX5600, and SRX5800 devices, IKE negotiations involving NAT traversal do not work if the IKE peer is behind a NAT device that will change the source IP address of the IKE packets during the negotiation. For example, if the NAT device is configured with DIP, it changes the source IP because the IKE protocol switches the UDP port from 500 to 4500.

The I/O cards (IOCs) and Services Processing Cards (SPCs) on SRX5000 line devices contain processing units that process a packet as it traverses the device. An IOC has one or more Network Processing Units (NPUs) and one or more Services Processing Units (SPUs) run on an SPC.

These processing units have different responsibilities. All flow-based services for a packet are executed on a single SPU. The responsibilities of these NPUs are not clearly delineated in regard to the other kind of services that run on them. .)

For example:

- An NPU processes packets discretely. It performs sanity checks and applies some screens that are configured for the interface, such as denial-of-service (DoS) screens, to the packet.
- An SPU manages the session for the packet flow and applies security features and other services to the packet. It also applies packet-based stateless firewall filters, classifiers, and traffic shapers to the packet.

- An NPU forwards a packet to the SPU using the hash algorithm. However, for some applications, like ALG, the system will need to query the application central point to determine on which SPU the packet should be processed.

These discrete, cooperating parts of the system, including the central point, each store the information identifying whether a session exists for a stream of packets and the information against which a packet is matched to determine if it belongs to an existing session.

This architecture allows the device to distribute processing of all sessions across multiple SPUs. It also allows an NPU to determine if a session exists for a packet, to check the packet, and to apply screens to the packet. How a packet is handled depends on whether it is the first packet in a flow.

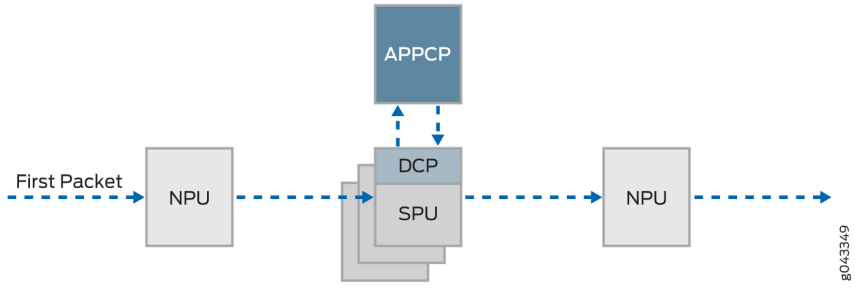
The following sections describe the processing architecture using SRX5600 and SRX5800 devices as an example:

Understanding First-Packet Processing

If the packet matches an existing flow, processing for the packet is assessed in the context of its flow state. The SPU maintains the state for each session, and the settings are then applied to the rest of the packets in the flow. If the packet does not match an existing flow, it is used to create a flow state and a session is allocated for it.

Figure 4 on page 42 illustrates the path the first packet in a flow takes as it enters the device—the NPU determines that no session exists for the packet, and the NPU sends the packet to the distributed central point to set up a distributed central point session. The distributed central point then sends a message to the application central point to select the SPU to set up a session for the packet and to process the packet. The distributed central point then sends the packet to that SPU. The SPU processes the packet and sends it to the NPU for transmission from the device. (This high-level description does not address application of features to a packet.)

Figure 4: First-Packet Processing

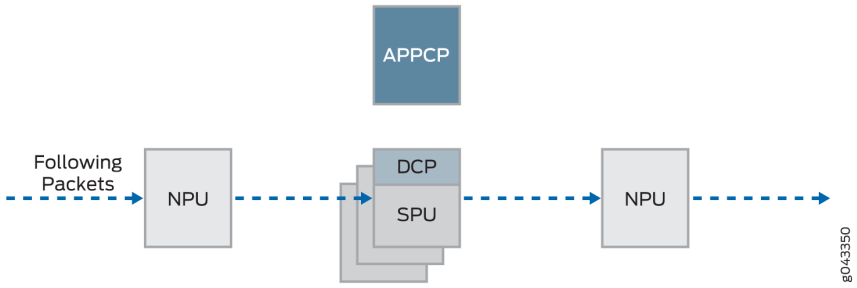


After the first packet in a flow has traversed the system and a session has been established for it, it undergoes fast-path processing.

Subsequent packets in the flow also undergo fast-path processing; in this case, after each packet enters the session and the NPU finds a match for it in its session table, the NPU forwards the packet to the SPU that manages its session.

Figure 5 on page 43 illustrates fast-path processing. This is the path a packet takes when a flow has already been established for its related packets. (It is also the path that the first packet in a flow takes after the session for the flow that the packet initiated has been set up.) After the packet enters the device, the NPU finds a match for the packet in its session table, and it forwards the packet to the SPU that manages the packet's session. Note that the packet bypasses interaction with the central point.

Figure 5: Fast-Path Processing



The following section explains how a session is created and the process a packet undergoes as it transits the device.

Understanding Fast-Path Processing

Here is an overview of the main components involved in setting up a session for a packet and processing packets both discretely and as part of a flow as they transit the SRX5600 and SRX5800 devices:

- Network Processing Units (NPUs)—NPUs reside on IOCs. They handle packet sanity checking and application of some screens. NPUs maintain session tables that they use to determine if a session exists for an incoming packet or for reverse traffic.

  The NPU session table contains an entry for a session if the session is established on an SPU for a packet that had previously entered the device via the interface and was processed by this NPU. The SPU installs the session in the NPU table when it creates the session.

  An NPU determines if a session exists for a packet by checking the packet information against its session table. If the packet matches an existing session, the NPU sends the packet and the metadata for it to the SPU. If there is no session, the NPUs sends the packet to one SPU which is calculated using the hash algorithm.

- Services Processing Units (SPUs)—The main processors of the SRX5600 and SRX5800 devices reside on SPCs. SPUs establish and manage traffic flows and perform most of the packet processing on a packet as it transits the device. Each SPU maintains a hash table for fast session lookup. The SPU applies stateless firewall filters, classifiers, and traffic shapers to traffic. An SPU performs all flow-based processing for

a packet and most packet-based processing. Each multicore SPU processes packets independently with minimum interaction among SPUs on the same or different SPC. All packets that belong to the same flow are processed by the same SPU.

The SPU maintains a session table with entries for all sessions that it established and whose packets it processes. When an SPU receives a packet from an NPU, it checks its session table to ensure that the packet belongs to it. It also checks its session table when it receives a packet from the distributed central point and sends a message to establish a session for that packet to verify that there is not an existing session for the packet.

- Central point—The central point architecture is divided into two modules, the application central point and the distributed central point. The application central point is responsible for global resource management and loading balancing, while the distributed central point is responsible for traffic identification (global session matching). The application central point functionality runs on the dedicated central point SPU, while the distributed central point functionality is distributed to the rest of the SPUs. Now the central point sessions are no longer on the dedicated central point SPU, but with the distributed central point on other flow SPUs.
- Routing Engine—The Routing Engine runs the control plane.

Understanding the Data Path for Unicast Sessions

This section describes the process of establishing a session for packets belonging to a flow that transits the device.

To illustrate session establishment and the packet "walk" including the points at which services are applied to the packets in a flow, this example uses the simple case of a unicast session.

This packet "walk" brings together the packet-based processing and flow-based processing that Junos OS performs on the packet.

Session Lookup and Packet-Match Criteria

To determine if a packet belongs to an existing flow, the device attempts to match the packet's information to that of an existing session based on the following six match criteria:

- Source address
- Destination address
- Source port
- Destination port
- Protocol
- Unique token from a given zone and virtual router

Understanding Session Creation: First-Packet Processing

This section explains how a session is set up to process the packets composing a flow. To illustrate the process, this section uses an example with a source “a” and a destination “b”. The direction from source to destination for the packets of the flow is referred to as (a ->b). The direction from destination to source is referred to as (b->a).

Step 1. A Packet Arrives at an Interface on the Device And the NPU Processes It.

This section describes how a packet is handled when it arrives at an SRX Series device ingress IOC.

1. The packet arrives at the device’s IOC and is processed by the NPU on the IOC.
2. The NPU performs basic sanity checks on the packet and applies some screens configured for the interface to the packet.
3. The NPU checks its session table for an existing session for the packet. (It checks the packet’s tuple against those of packets for existing sessions in its session table.)
    a. If no existing session is found, the NPU forwards the packet to the hash SPU.
    b. If a session match is found, the session has already been created on an SPU that was assigned to it, so the NPU forwards the packet to the SPU for processing along with the session ID.

Example: Packet (a ->b) arrives at NPU1. NPU1 performs sanity checks and applies DoS screens to the packet. NPU1 checks its session table for a tuple match, and no existing session is found. NPU1 forwards the packet to an SPU.

Step 2. The Distributed Central Point Creates a Session with a "Pending” State.

When an NPU receives a packet, the NPU send it to the distributed central point, based on the hash algorithm. The distributed central point then looks up the distributed central point session table and creates an entry if needed.

This process entails the following parts:

1. The distributed central point checks its session table to determine if a session exists for the packet received from the NPU. (An NPU forwards a packet to the distributed central point because it cannot find an existing session for the packet)
2. If there is no entry that matches the packet in the distributed central point session table, the distributed central point creates a pending wing for the session. The distributed central point then sends a query message to the application central point to select an SPU to be used for the session.
3. On receiving the query message, the application central point checks its gate table to determine if a gate exists for the packet. If a gate is matched or some other session distribution algorithm is triggered, the application central point selects another SPU to process the packet; otherwise, the SPU (that is, the distributed central point SPU) is selected. Finally, the application central point sends a query response to the distributed central point.
4. On receiving the query response, the distributed central point forwards the first packet in flow to the selected SPU in a message directing the SPU to set up a session locally to be used for the packet flow.

For example, the distributed central point creates a pending wing (a ->b) for the session. The application central point selects SPU1 to be used for it. The distributed central point sends SPU1 the (a->b) packet along with a message to create a session for the distributed central point.

Example: The distributed central point creates a pending wing (a ->b) for the session. It selects SPU1 to be used for it. It sends SPU1 the (a->b) packet along with a message to create a session for it.

Step 3. The SPU Sets Up the Session.

Each SPU, too, has a session table, which contains information about its sessions. When the SPU receives a message from the distributed central point to set up a session, it checks its session table to ensure that a session does not already exist for the packet.

1. If there is no existing session for the packet, the SPU sets up the session locally.
2. The SPU sends a message to the distributed central point directing it to install the session.

> **NOTE:** During first-packet processing, if NAT is enabled, the SPU allocates IP address resources for NAT. In this case, the first-packet processing for the session is suspended until the NAT allocation process is completed.

The SPU adds to the queue any additional packets for the flow that it might receive until the session has been installed.

Example: SPU1 creates the session for (a ->b) and sends a message back to the distributed central point directing it to install the pending session.

Step 4. The Distributed Central Point Installs the Session.

The distributed central point receives the install message from the SPU.

1. The distributed central point sets the state for the session's pending wing to active.
2. The distributed central point installs the reverse wing for the session as an active wing.

> **NOTE:** For some cases, such as NAT, the reverse wing may be installed on a different distributed central point from the init wing distributed central point.

3. It sends an acknowledge (ACK) message to the SPU, indicating that the session is installed.

Example: The distributed central point receives a message from SPU1 to install the session for the (a->b) wing. It sets the session state for the (a->b) wing to active. It installs the reverse wing (b->a) for the session and makes it active; this allows for delivery of packets from the reverse direction of the flow: destination (b) to be delivered to the source (a).

Step 5. The SPU Sets Up the Session on the Ingress and Egress NPUs.

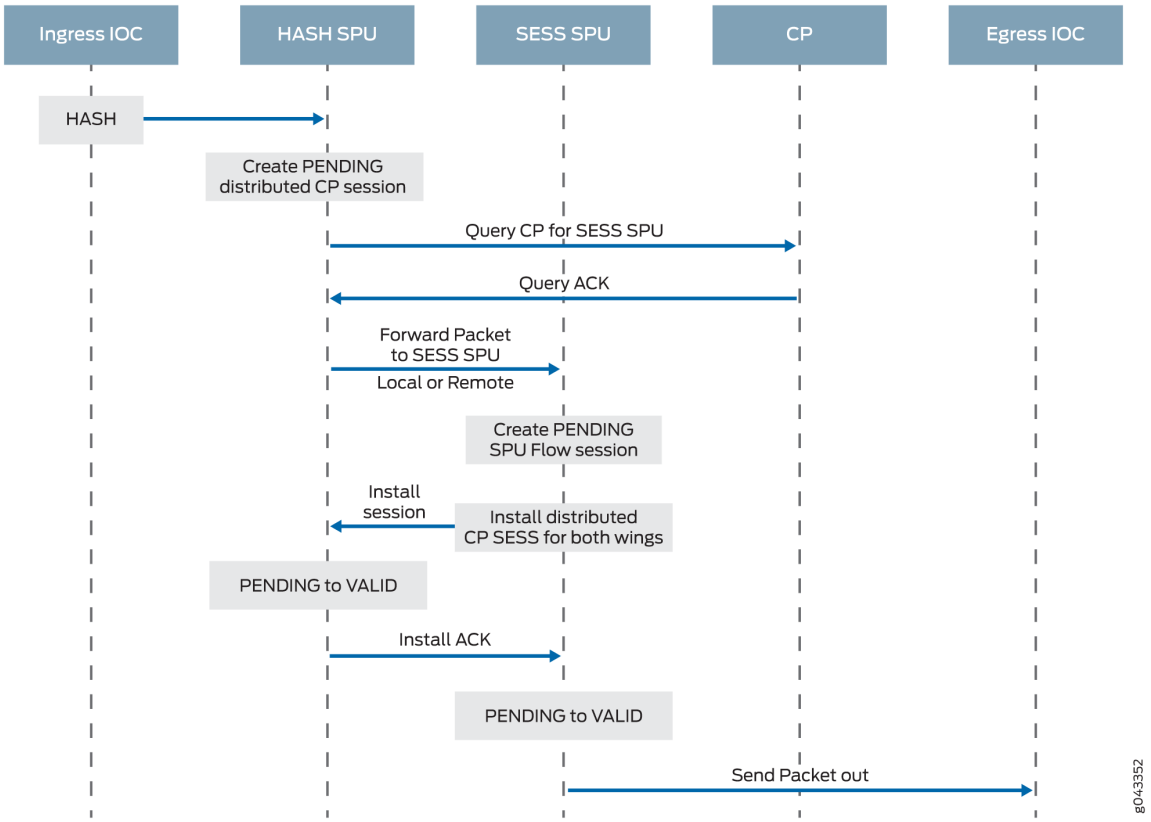
NPUs maintain information about a session for packet forwarding and delivery. Session information is set up on the egress and ingress NPUs (which sometimes are the same) so that packets can be sent directly to the SPU that manages their flows and not to the distributed central point for redirection.

Step 6. Fast-Path Processing Takes Place.

For the remainder of the steps entailed in packet processing, proceed to Step 1 in "Understanding Fast-Path Processing" on page 47.

Figure 6 on page 47 illustrates the first part of the process that the first packet in a flow undergoes after it reaches the device. At this point a session is set up to process the packet and the rest of the packets belonging to its flow. Subsequently, it and the rest of the packets in the flow undergo fast-path processing.

Figure 6: Session Creation: First-Packet Processing



Understanding Fast-Path Processing

All packets undergo fast-path processing. However, if a session exists for a packet, the packet undergoes fast-path processing and bypasses the first-packet process. When there is already a session for the packet's flow, the packet does not transit the central point.

Here is how fast-path processing works: NPUs at the egress and ingress interfaces contain session tables that include the identification of the SPU that manages a packet's flow. Because the NPUs have this session information, all traffic for the flow, including reverse traffic, is sent directly to that SPU for processing.

To illustrate the fast-path process, this section uses an example with a source "a" and a destination "b". The direction from source to destination for the packets of the flow is referred to as (a->b). The direction from destination to source is referred to as (b->a).

Step 1. A Packet Arrives at the Device and the NPU Processes It.

This section describes how a packet is handled when it arrives at a services gateway's IOC.

1. The packet arrives at the device's IOC and is processed by the NPU on the card.

   The NPU performs sanity checks and applies some screens, such as denial-of-service (DoS) screens, to the packet.

2. The NPU identifies an entry for an existing session in its session table that the packet matches.
3. The NPU forwards the packet along with metadata from its session table, including the session ID and packet tuple information, to the SPU that manages the session for the flow, applies stateless firewall filters and CoS features to its packets, and handles the packet's flow processing and application of security and other features.

Example: Packet (a ->b) arrives at NPU1. NPU1 performs sanity checks on the packet, applies DoS screens to it, and checks its session table for a tuple match. It finds a match and that a session exists for the packet on SPU1. NPU1 forwards the packet to SPU1 for processing.

Step 2. The SPU for the Session Processes the Packet.

Most of a packet's processing occurs on the SPU to which its session is assigned. The packet is processed for packet-based features such as stateless firewall filters, traffic shapers, and classifiers, if applicable. Configured flow-based security and related services such as firewall features, NAT, ALGs, and so on, are applied to the packet. (For information on how security services are determined for a session.

1. Before it processes the packet, the SPU checks its session table to verify that the packet belongs to one of its sessions.
2. The SPU processes the packet for applicable features and services.

Example: SPU1 receives packet (a->b) from NPU1. SPU1 checks its session table to verify that the packet belongs to one of its sessions. Then it processes packet (a ->b) according to input filters and CoS features that apply to its input interface. The SPU applies the security features and services that are configured for the packet's flow to it, based on its zone and policies. If any are configured, it applies output filters, traffic shapers and additional screens to the packet.

Step 3. The SPU Forwards the Packet to the NPU.

1. The SPU forwards the packet to the NPU.
2. The NPU applies any applicable screens associated with the interface to the packet.

Example: SPU1 forwards packet (a ->b) to NPU2, and NPU2 applies DoS screens.

Step 4. The Interface Transmits the Packet from the Device.

Example: The interface transmits packet (a->b) from the device.

Step 5. A Reverse Traffic Packet Arrives at the Egress Interface and the NPU Processes It.

This step mirrors Step 1 exactly in reverse. See Step 1 in this section for details.

Example: Packet (b->a) arrives at NPU2. NPU2 checks its session table for a tuple match. It finds a match and that a session exists for the packet on SPU1. NPU2 forwards the packet to SPU1 for processing.

Step 6. The SPU for the Session Processes the Reverse Traffic Packet.

This step is the same as Step 2 except that it applies to reverse traffic. See Step 2 in this section for details.

Example: SPU1 receives packet (b->a) from NPU2. It checks its session table to verify that the packet belongs to the session identified by NPU2. Then it applies packet-based features configured for the NPU1's interface to the packet. It processes packet (b->a) according to the security features and other services that are configured for its flow, based on its zone and policies.

Step 7. The SPU Forwards the Reverse Traffic Packet to the NPU.

This step is the same as Step 3 except that it applies to reverse traffic. See Step 3 in this section for details.

Example: SPU1 forwards packet (b->a) to NPU1. NPU1 processes any screens configured for the interface.

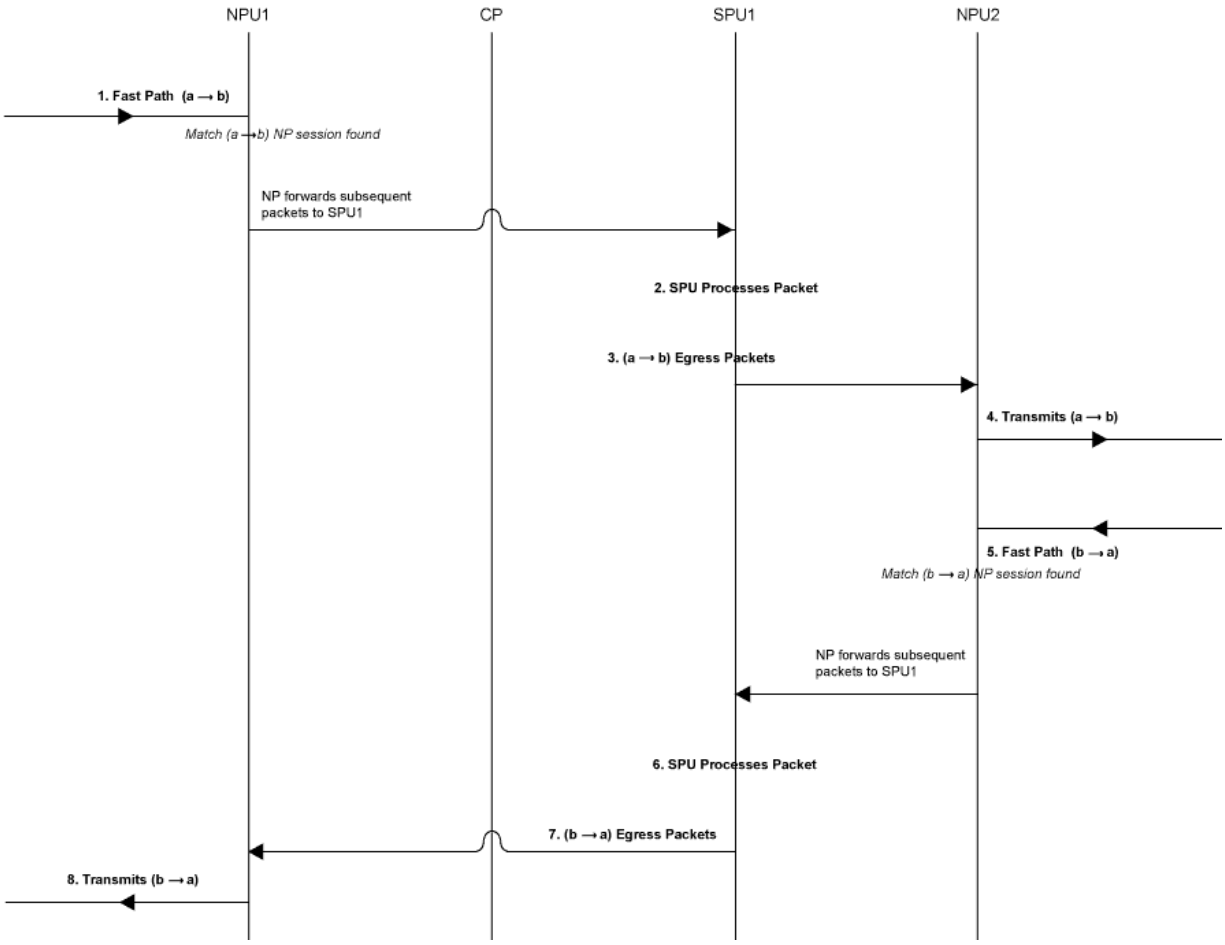
8. The Interface Transmits the Packet from the Device.

This step is the same as Step 4 except that it applies to reverse traffic. See Step 4 in this section for details.

Example: The interface transmits packet (b->a) from the device.

Figure 7 on page 50 illustrates the process a packet undergoes when it reaches the device and a session exists for the flow that the packet belongs to.

Figure 7: Packet Walk for Fast-Path Processing



Understanding Services Processing Units

For a given physical interface, the SPU receives ingress packets from all network processors in the network processor bundle associated with the physical interface. The SPU extracts network processor bundle information from the physical interface and uses the same 5-tuple hash algorithm to map a flow to a network processor index. To determine the network processor, the SPU does a lookup on the network processor index in the network processor bundle. The SPU sends egress packets to the physical interface's local Physical Interface Module (PIM) for the outward traffic.

NOTE: The network processor and the SPU use the same 5-tuple hash algorithm to get the hash values for the packets.

Understanding Scheduler Characteristics

For SRX5600 and SRX5800 devices, the IOC supports the following hierarchical scheduler characteristics:

- IFL – The configuration of the network processor bundle is stored in the physical interface data structure. For example, SRX5600 and SRX5800 devices have a maximum of 48 PIMs. The physical interface can use a 48-bit bit-mask to indicate the PIM, or the network processor traffic from this physical interface is distributed in addition to the physical interface's primary network processor.

  On SRX5000 line devices, the iflset functionality is not supported for aggregated interfaces like *reth*.

- IFD – The logical interface associated with the physical interface of a network processor bundle is passed to all the IOCs that have a PIM in the network processor bundle.

Understanding Network Processor Bundling

The network processor bundling feature is available on SRX5000 line devices. This feature enables distribution of data traffic from one interface to multiple network processors for packet processing. A primary network processor is assigned for an interface that receives the ingress traffic and distributes the packets to several other secondary network processors. A single network processor can act as a primary network processor or as a secondary network processor to multiple interfaces. A single network processor can join only one network processor bundle.

Network Processor Bundling Limitations

Network processor bundling functionality has the following limitations:

- Network processor bundling allows a total of 16 PIMs per bundle and 8 different network processor bundle systems.
- You need to reboot the device to apply the configuration changes on the bundle.
- Network processor bundling is below the reth interface in the overall architecture. You can choose one or both interfaces from the network processor bundle to form the reth interface.
- If the IOC is removed from a network processor bundle, the packets forwarded to the PIM on that IOC are lost.
- When the network processor bundle is enabled, the ICMP, UDP, and TCP sync flooding thresholds no longer apply to an interface. Packets are distributed to multiple network processors for processing. These thresholds apply to each network processor in the network processor bundle.
- Network processor bundling is not supported in Layer 2 mode.
- Because of memory constraints on the network processor, the number of network processor bundled ports that are supported per PIM is limited. Within the network processor bundle, each port needs to have a global port index. The global port index is calculated using the following formula:

Global_port_index = (global_pic * 16) + port_offset

- Link aggregation groups (LAGs) and redundant Ethernet interface LAGs in chassis cluster implementations can coexist with network processor bundling. However, neither LAGs nor redundant Ethernet interface LAGs can overlap with or share physical links with a network processor bundle.

Configuring IOC to NPC Mapping

An Input/Output card (IOC) to Network Processing Card (NPC) mapping requires you to map one IOC to one NPC. However, you can map multiple IOCs to a single NPC. To balance the processing power in the NPC on the SRX3400, SRX3600, and SRX4600 Services Gateways, the chassis process (daemon) runs an algorithm that performs the mapping. It maps an IOC to an NPC that has the least amount of IOCs mapped to it. You can also use the command-line interface (CLI) to assign a specific IOC to a specific NPC. When you configure the mapping, the chassis process will first use your configuration, then apply the least-number NPC algorithm for the rest of the IOCs.

NOTE: Platform support depends on the Junos OS release in your installation.

To configure the IOC to NPC mapping:

```
[edit]
set chassis ioc-npc-connectivity {
    ioc slot-number npc (none | slot-number);
}
```

See Table 3 on page 52 for a description of the **set chassis ioc-npc-connectivity** options.

Table 3: IOC to NPC Connectivity Options

Option	Description
ioc *slot-number*	Specify the IOC slot number. Range is 0 through 7 for SRX3400 devices and 0 through 12 for SRX3600 devices.
npc *slot-number*	Specify the NPC slot number. Range is 0 through 7 for SRX3400 devices and 0 through 12 for SRX3600 and SRX 4600 devices.
none	The chassis process maps the connection for the particular IOC.

> **NOTE:** You must restart the chassis control after you commit the **set chassis ioc-npc-connectivity** command.

Understanding Flow Processing on SRX5K-SPC3 Devices

The service processing card SRX5K-SPC3 is introduced to improve the performance of security services on the SRX5000 security services gateway. The SPC3 card supports higher throughput, maintains its reliability as it preserves the chassis cluster functionality and scalability for service processing.

The SPC3 card provides support for the following security features:

- Application layer gateway (ALG). [See ALG Overview]
- Advanced anti-malware (Juniper Sky ATP). [See Juniper Sky Advanced Threat Prevention Administration]
- Application security suite. [See Application Security User Guide for Security Devices]
- Flow-based packet processing implementation
- GPRS tunneling protocol (GTP) and stream control transmission protocol (SCTP). [See General Packet Radio Service User Guide for Security Devices]
- High availability (chassis cluster). [See Chassis Cluster User Guide for SRX Series Devices]
- Intrusion detection and prevention (IDP). [See Intrusion Detection and Prevention Overview]
- Network address translation (NAT). [See Network Address Translation User Guide for Security Devices]
- Stateful firewall
- SSL proxy. [See SSL Proxy]
- Firewall user authentication. [See Authentication and Integrated User Firewalls User Guide for Security Devices]
- UTM (antivirus, web filtering, content filtering, and antispam). [See UTM User Guide for Security Devices]

The security flow is enhanced to support SPC3 card with all the existing security features that are supported on the SPC2 card.

NOTE:
The following limitations apply for the SPC3 card in Junos OS Release 18.2R1-S1:

- Interoperability of SPC3 card and SPC2 card is not supported.
- IPsec VPN functionality is not supported with SPC3 card.

Starting in Junos OS Release 18.2R1-S1, a new service processing card (SPC3) is introduced for the SRX5000 Series devices. The introduction of the new card improves the scalability and performance of the device and maintains its reliability as it preserves the chassis cluster functionality. The SPC3 card supports higher throughput and scalability for service processing.

On SRX5000 Series devices, SPC3 card interoperates with I/O cards (IOC2, IOC3), Switch Control Board (SCB2, SCB3), Routing Engines and SPC2 cards.

Starting in Junos OS Release 18.4R1, a mix of of SPC3 and SPC2 cards is supported on SRX5000 Series devices.

If you are adding the SPC3 cards on SRX5000 line of devices, the new SPC3 card must installed in the lowest-numbered slot of any SPC. The SPC3 card is installed in the original lowest-numbered slot provides the central point (CP) functionality in mixed-mode.
For example, if your services gateway contains a mix of SPC2 and SPC3 cards, an SPC3 must occupy the lowest-numbered slot of any SPC in the chassis. This configuration ensures that the central point (CP) functionality in mixed-mode is performed by the SPC3 card.

On SRX5000 Series devices operating in mixed-mode, flow processing is shared between SPC3 and SPC2 cards. Central Point processing takes place on the lowest number SPC slot for which an SPC3 card is installed.

NOTE: When SRX Series devices are operating in a chassis cluster mode, SPC3 and SPC2 cards must be installed in the same slot locations on each chassis.

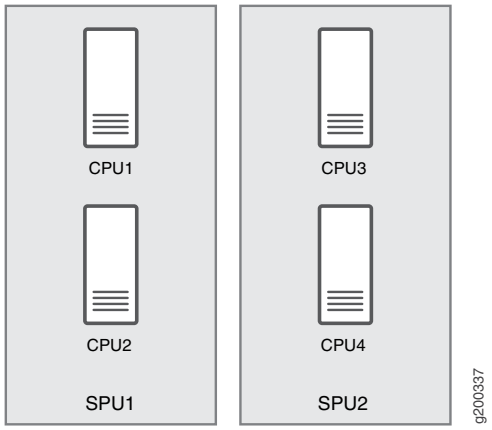
Understanding SPC3 Software Architecture

SPC3 flow architecture is same as CP-Lite architecture. The SPC3 physically has two Services Processing Units (SPU) and each SPU has two CPUs.

When you install one or two SPC3s, traffic processing utilizes 75% of the first SPC. When you install three or more SPC3s, the traffic processing utilizes 50% of the first SPC.

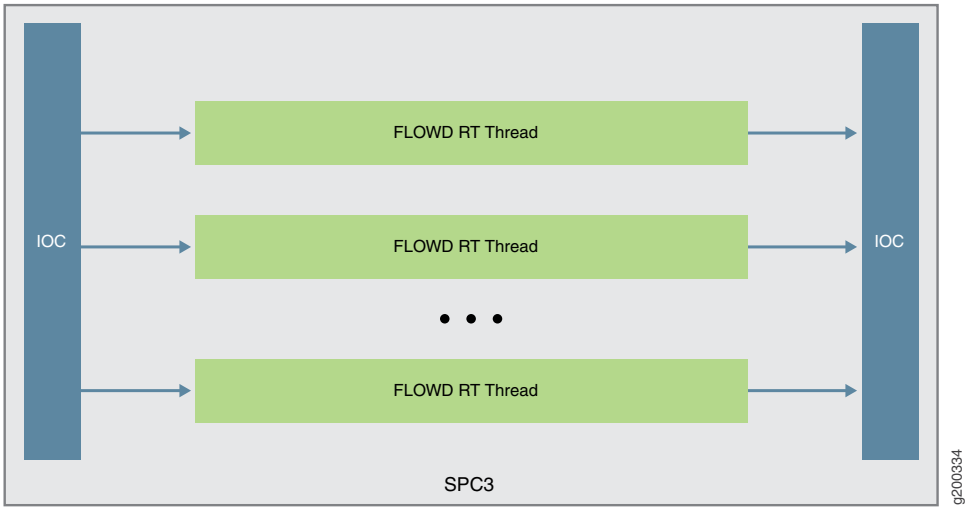
The way the IOC hashes the packets to process the flow is changed. Figure shows the packet flow of SRX device with SPC3.

Figure 8: Packet flow on SPC3



On SPC3, packets are distributed from IOC to each core directly. Since the IOC directly hashes packets to the flowd RT thread, the original LBT thread is removed. The packets are now delivered to the flowd thread instead of SPU. If the security flow installs NP sessions, instead of SPU ID, the session thread ID is used by IOC to forward packets to correct thread associate with the session.

Figure 9: Packet flow through flowd thread



Understanding Load Distribution

All packets that come through a revenue port will be distributed to different SPUs based on hash algorithm, which is same as the existing SRX5000 Line devices hash based on CP-Lite architecture. The hash method varies for different types of traffic. The table below lists hash methods.

Table 4: Load Distribution - Hash Methods

Protocol		Ports	Hash Method
TCP		L4 src port and dst port	Hashed by 5-tuple
UDP	Normal	L4 src port and dst port	Hashed by 5-tuple
	GTP	L4 src port and dst port	Hashed by 5-tuple
	IKE	L4 src port and dst port	Hashed by IP pair
ICMP		1. ICMP version 4 info message ICMP_ECHO/ICM_ECHOREPLY id/seq ICMP_TSTAMP/ICMP_TSTAMPREPLY id/seq ICMP_IREQ/ICMP_IREQREPLY id/seq ICMP_MASKREQ/ICMP_MASKREPLY 0x00010001 2. ICMP version 6 info message ICMP6_ECHO_REPLY/ICMP6_ECHO_REQUEST id/seq 3. ICMP error message Match by embedded IP 4. All others 0x00010001	ICMP info is hashed by 5-tuple; ICMP error is hashed by 3-tuple (no ports info)
SCTP		L4 src port and dst port	Hashed by 5-tuple
ESP		SPI	Hashed by IP pair
AH		SPI	Hashed by IP pair
GRE		If PPTP alg is enabled, sport = call id; dport = 0 By default, port is 0x00010001	Hashed by 3-tuple
PIM		By default, PIM ports 0x00010001	Hashed by 3-tuple
FRAGMENT		First fragment, has the normal ports None first fragment, no ports	Hashed by 3-tuple
Other IP packet		Ports 0x00010001	Hashed by 3-tuple

Table 4: Load Distribution - Hash Methods *(continued)*

Protocol	Ports	Hash Method
NONE IP	Not applicable	Hashed by Mac address and Ethernet Type (Vlan ID)

Understanding NP Session and Service Offload (SOF)

Network processor (NP) session is an IOC-based session that allows and establishes the SPU sessions. The packets that pass the NP session has the following advantages:

- Avoids session lookup on SPU to gain better performance.
- Avoids extra packet forwarding between session SPU and hash SPU.

Service offload is a special type of NP session to provide low-latency feature for session that needs basic firewall service. Packets that hits the SOF session on an IOC bypass the packet processing on SPU and is directly forwarded by IOC. The following traffic types support service offload:

- Basic firewall (without plugin and fragments), IPv4 and IPv6 TCP, UDP traffic
- IPv4 NAT
- 1Fan-in and 1Fan-out Multicast
- ALGs such as FTP data session

Understanding J-Flow support on SPC3

J-Flow is the juniper version of industry standard traffic monitoring mechanism. It provides a feature to export snapshot of network traffic statistics to the remote server for network monitoring and further data processing. J-Flow supports v5, v8 and v9 format. All these three versions are supported on SPC3.

Understanding Datapath Debug SPU Support (E2E)

Datapath debug provides filter based end-to-end (E2E) packet debug feature on SRX5000 Line devices. It traces packet path and dump packet content.

On SPC3, JEXEC is the only E2E event type that is supported and the following E2E action types are supported:

- Count
- Dump

- Trace
- Trace-summary

Understanding Fragmentation Handling, ISSU, and ISHU Support

On SPC3, fragmented packets are forwarded to “fragment core” in a specific PFE based on its header tuple values. After receiving a fragmented packet, flow performs defragmentation and forwards the packet to its session core. The flow logic does not change and remains the same.

While performing the ISSU, the virtual SPUs are synchronized to related virtual SPU IDs. The ISHU support is based on CP-Lite architecture. Basically, two ISHU operations are supported:

- Insert a new SPC to secondary node.
- Replace an SPC on secondary node, and the number of SPCs should be same as that of primary node.

RELATED DOCUMENTATION

Understanding SRX Series Services Gateways Central Point Architecture | 59

Configuring Inline J-Flow to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices

Release History Table

Release	Description
18.4R1	Starting in Junos OS Release 18.4R1, a mix of of SPC3 and SPC2 cards is supported on SRX5000 Series devices.
18.2R1-S1	Starting in Junos OS Release 18.2R1-S1, a new service processing card (SPC3) is introduced for the SRX5000 Series devices. The introduction of the new card improves the scalability and performance of the device and maintains its reliability as it preserves the chassis cluster functionality. The SPC3 card supports higher throughput and scalability for service processing.
15.1X49-D70	By default, the SRX Series device is enabled for flow-based forwarding for IPv4 traffic on all devices, apart from the SRX300 Series and SRX550M devices that are set to drop mode. Starting with Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, for the SRX1500 series, SRX4100, SRX4200, SRX5400, SRX5600, SRX5800 and vSRX devices, you do *not* need to reboot the device when you are switching modes between flow mode, packet mode, and drop mode. For SRX300 Series and SRX550M devices, you *must* reboot the device when switching between flow mode, packet mode, and drop mode.

RELATED DOCUMENTATION

Flow-Based Sessions | 68

Central Point Architecture in Security Devices Overview

IN THIS SECTION

- Understanding SRX Series Services Gateways Central Point Architecture | 59
- Understanding Enhancements to Central Point Architecture for the SRX5000 Line | 62
- Understanding Central Point Architecture Flow Support for GTP and SCTP | 64

The central point delegates the session processing to one of the SPUs. When a session is not established, the central point selects an SPU to establish the session for the flow, based on load- balancing criteria. If the session already exists, the central point forwards packets for that flow to the SPU hosting it.

Understanding SRX Series Services Gateways Central Point Architecture

IN THIS SECTION

- Load Distribution in mixed Mode | 61
- Sharing Processing Power and Memory in mixed Mode | 61

The central point (CP) architecture has two basic flow functionalities: load balancing and traffic identification (global session matching). As described in this topic, the central point architecture is implemented either in centric mode, in which all session distribution and session matching is performed by the central point, or in mixed-mode, in which a percentage of Services Processing Unit (SPU) is dedicated to performing the central point functionality.

The central point's main function is to delegate session processing to one of the SPUs. If the session has not yet been established, the central point selects an SPU to establish the session for the flow, based on load- balancing criteria. If the session already exists, the central point forwards packets for that flow to the SPU hosting it. It also redirects packets to the correct SPU in the event that the NPU fails to do so.

The central point maintains a global session table with information about the owner SPU of a particular session. It functions as a central repository and resource manager for the whole system.

> **NOTE:** The central point architecture is also implemented in CP-lite mode in which session management is offloaded from the central point to SPUs for performance and session scaling improvement. CP-lite is not discussed in this topic.

The SRX Series device type in conjunction with the Junos OS release determine which mode is supported.

Table 5 on page 60 identifies the central point architecture implementation that is supported on SRX Series devices for various releases.

Table 5: Central Point Implementation on SRX Series Devices in Conjunction With Junos OS Releases

	Mode Supported on SRX1400	Mode Supported On SRX3000 Series Devices	Mode Supported on SRX5000 Series Devices
Junos OS Release 12.3X48 and Previous Releases	• CP centric • mixed-mode	• CP centric • mixed-mode	• CP centric • mixed-mode
• **Junos OS Release 15.1X49-D10** • **Junos OS Release 15.1X49-D15** • **Junos OS Release 15.1X49-D20**	These SRX Series devices are no longer supported.	These SRX Series devices are no longer supported.	• CP centric **NOTE:** NG-SPC renders combo mode obsolete.
Junos OS Release 15.1X49-D30 and later releases	These SRX Series devices are no longer supported.	These SRX Series devices are no longer supported.	• CP-lite **NOTE:** NG-SPC renders mixed-mode obsolete.

The central point forwards a packet to its Services Processing Unit (SPU) upon session matching, or distributes traffic to an SPU for security processing if the packet does not match any existing session. The central point architecture is implemented in CP centric mode, in which all session distribution and session matching is performed by the CP or in combo mode

On some SRX Series devices, an entire SPU cannot be dedicated for central point functionality, but a certain percentage of the SPU is automatically allocated for central point functionality and the rest is

allocated for normal flow processing. When an SPU performs the function of central point as well as normal flow processing, it is said to be in combination, or *mixed*, mode.

The percentage of SPU dedicated to the central point functionality depends on the number of SPUs in the device. Based on the number of SPUs, there are three modes available on the SRX Series devices— small central point, medium central point, and large central point.

In small central point mode, a small percentage of an SPU is dedicated to central point functionality and the rest is dedicated to the normal flow processing. In medium central point mode, an SPU is almost equally shared for central point functionality and normal flow processing. In large central point mode, an entire SPU is dedicated to central point functionality. In mixed-mode, the central point and SPU share the same load-balancing thread (LBT) and packet-ordering thread (POT) infrastructure.

This topic includes the following sections:

Load Distribution in mixed Mode

The central point maintains SPU mapping table (for load distribution) that lists live SPUs with the logic SPU IDs mapped to the physical Trivial Network Protocol (TNP) addresses mapping. In mixed-mode, the SPU that hosts the central point is included in the table. The load distribution algorithm is adjusted based on session capacity and processing power to avoid overloading of sessions.

Sharing Processing Power and Memory in mixed Mode

The CPU processing power in a mixed-mode SPU is shared based on the platform and the number of SPUs in the system. Similarly, the CPU memory is also shared between the central point and SPU.

An SPU has multiple cores (CPUs) for networking processing. In "small" SPU mixed-mode, CPU functionality takes a small portion of the cores, whereas "medium" SPU mixed-mode requires a larger portion of cores. The processing power for central point functionalities and flow processing is shared, based on the number of Services Processing Cards (SPC), as shown in Table 6 on page 61. Platform support depends on the Junos OS release in your installation.

Table 6: mixed Mode Processing

SRX Series device	Central point mode with 1 SPC or SPC2	Central point mode with 2 or more SPCs or SPC2s	Central point mode with 1 or 2 SPC3s	Central point mode with more than 2 SPC3s
SRX1400	Small	Medium	NA	NA
SRX3400	Small	Medium	NA	NA
SRX3600	Small	Medium	NA	NA

Table 6: mixed Mode Processing *(continued)*

SRX Series device	Central point mode with 1 SPC or SPC2	Central point mode with 2 or more SPCs or SPC2s	Central point mode with 1 or 2 SPC3s	Central point mode with more than 2 SPC3s
SRX3400 (expanded performance and capacity license)	Small	Large	NA	NA
SRX3600 (expanded performance and capacity license)	Small	Large	NA	NA
SRX5600	Large	Large	Medium	Large
SRX5800	Large	Large	Medium	Large
SRX5400	Large	Large	Medium	Large

NOTE: The mixed-mode processing only exists with SPCI on SRX1400, SRX3400, SRX3600, and SRX5000 line devices.

Understanding Enhancements to Central Point Architecture for the SRX5000 Line

Previously, for the SRX5000 line of services gateways, the central point was a bottleneck in device performance and scaling. When more Services Processing Cards (SPCs) were integrated into the system, the overall processing power increased linearly, but the system connections per second (cps) remained constant and could not be improved because of the single centralized point in the system. This severely impacted the overall system utilizations in both capacity and cps.

Starting with Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, on SRX5000 line devices, the central point architecture is enhanced to handle higher connections per second (cps). The new central point architecture prevents data packets from going through the central point by off-loading session management functionalists to the Services Processing Unit (SPU). Therefore, data packets are directly forwarded from the network processing unit to the SPU instead of going through the central point.

The central point architecture is divided into two modules, the application central point and the distributed central point. The application central point is responsible for global resource management and loading balancing, while the distributed central point is responsible for traffic identification (global session matching). The application central point functionality runs on the dedicated central point SPU, while the distributed central point functionality is distributed to the rest of the SPUs. Now the central point sessions are no longer on the dedicated central point SPU, but with distributed central point on other flow SPUs.

NOTE: The central point for SRX5000 line refers to the application central point, or the distributed central point or both, with respect to global resource management and load balancing, it refers to the application central point, whereas with respect to traffic identification and session management, it refers to the distributed central point (sometimes referred to the SPU as well).

NOTE: The SNMP log and SNMP trap were generated by the central point with rate limit. Now, the SNMP log and SNMP trap are generated by the SPU or central point. As there is more than one SPU, the number of SNMP log and traps generated are more. To verify the number of connections per second (CPS) on the device run **SNMP MIB walk nxJsNodeSessionCreationPerSecond** command. The SNMP polling mechanism calculates the CPS value based on the average number of CPS in the past 96 seconds. So, if the CPS is not constant, the number of CPS reported is inaccurate.

Understanding Central Point Session Limit Performance Enhancements

Starting in Junos OS 15.1X49-D70 and Junos OS Release 17.3R1, a new session connection (conn-tag) tag option is available to allow you to add a flow filter to further distinguish GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.

The flow session connection tuple consists of a 32-bit connection tag that is used to uniquely identify GTP-U sessions and SCTP sessions that are not distinguishable by the six part tuple only. You can configure the system to include the session connection tag tuple to identify GTP-U sessions and SCTP sessions by adding the session connection tag to the standard six tuples that identify a session. The system determines the DCP for GTP-U/SCTP by hashing the session connection tag.

The central point architecture distributes GTP-U traffic handled by a gateway GPRS support node (GGSN) and SGSN pair on all SPUs by switching to tunnel endpoint identifier (TEID)-based hash distribution. To handle load-balancing issues, tag-based hash distribution is used to ensure even distribution of SCTP traffic from different associations among all SPUs. (The connection tag for GTP-U is the TEID and for SCTP is the vTag.)

Understanding Central Point Architecture Flow Support for GTP and SCTP

Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, the central point architecture provides enhanced support for GPRS tunneling protocol, control (GTP-C), GPRS tunneling protocol, user plane (GTP-U), and Stream Control Transmission Protocol (SCTP).

The central point architecture, which is supported on the SRX5400, SRX5600, and SRX5800 devices, is enhanced to address the GTP-C message rate-limiting to protect gateway GPRS support node (GGSN) from GTP-C message flood, to prevent GTP-C packet drop issues during SGSN handover, and to distribute GTP-U traffic handled by a GGSN and SGSN pair on all SPUs by switching to tunnel endpoint identifier (TEID)-based hash distribution. Use the **enable-gtpu-distribution** command to enable or disable GTP-U session distribution. By default, the **enable-gtpu-distribution** command is disabled.

Connection-tag to flow session tuple is introduced to resolve GTP/SCTP load balance issue. All session including Distributed CP (DCP) session and SPU session are modified to accommodate connection-tag. The session creation have following tuple: src-ip, dst-ip, src-port, dst-port, protocol, session-token and connection tag.

The GTP ALG requires GTP-C sessions to be fixed by hashing GGSN IP addresses. The GTP ALG deny GTP-C session creation if the first packet is of uncertain direction, which will cause packet drop. To prevent the GTP-C packets from being dropped, a new flow session is created and the GTP-C traffic is allowed to pass even if the GGSN or SGSN direction is not determined. Later, the GGSN IP is determined using the correct SPU to create the flow session and age out the old session. The intermittent packets hitting the old session will be forwarded to the new SPU and be processed on the new session.

To handle load-balancing issues, tag-based hash distribution is used to ensure even distribution of GTP-U/SCTP traffic among all SPUs. A 32-bit connection tag is introduced that uniquely identifies the GTP-U and the SCTP sessions. The connection tag for GTP-U is the TEID and for SCTP is the vTag. The default connection-tag is 0. The connection tag remains 0 if it is not used by the sessions. Flow will determine connection tag for GTP-U/SCTP sessions and distribute them by hashing connection tag.

A SCTP association is a connection between two SCTP endpoints. Each SCTP endpoint identifies the association with a tag. During association setup (4-way handshakes), two SCTP endpoints exchange their own tags for packet receiving. During 4-way handshake, the receiver of INIT/INIT-ACK records the value of itag, and places into the vtag field of every SCTP packet that transmit within this association. Then the peer uses the vtag to validate the sender of this packet.

Flow sessions created after CP-Lite as follows:

SPU is selected by hash(tag), the Client to Server traffic is handled on hash (tagB) SPU then forwarded to hash (tagA) SPU. Server to Client traffic is handled on hash (tagA) SPU directly.

1. After receive INIT packet, on hash (tagA) SPU:

   DCP-session A1: client=> server, SCTP, Conn ID: 0x0;

   Session A1: client=> server, SCTP, Conn ID: 0x0;

   On hash (tagB) SPU: no session.

2. After receive INIT-ACK packet, on hash (tagA) SPU:

   DCP-session A1: client=> server, SCTP, Conn ID: 0x0;

   DCP-session A2: server => client, SCTP, Conn ID: tagA;

   Session A1: client=> server, SCTP, Conn ID: 0x0;

   Session A2: server => client, SCTP, Conn ID: tagA;

   On hash (tagB) SPU: no session.

3. After receive COOKIE-ECHO packet, on hash (tagA) SPU:

   DCP-session A1: client=> server, SCTP, Conn ID: 0x0;

   DCP-session A2: server => client, SCTP, Conn ID: tagA;

   Session A1: client=> server, SCTP, Conn ID: 0x0;

   Session A2: server => client, SCTP, Conn ID: tagA;

   Session A3: client=> server, SCTP, Conn ID: tagB;

   On hash (tagB) SPU:

   DCP-session: client => server, SCTP, Conn ID: tag B

4. After receive COOKIE-ACK packet, flow sessions have no change.

5. After handshake succeeds, HEARBEAT will be send on all paths.

Understanding the Flow Session Connection Filter Option

Starting in Junos OS 15.1X49-D70 and Junos OS Release 17.3R1, a new session connection (conn-tag) tag option is available to allow you to add a flow filter to further distinguish GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.

The flow session connection tuple consists of a 32-bit connection tag that is used to uniquely identify GTP-U sessions and SCTP sessions that are not distinguishable by the six part tuple only. You can configure the system to include the session connection tag tuple to identify GTP-U sessions and SCTP sessions by adding the session connection tag to the standard six tuples that identify a session. The system determines the DCP for GTP-U/SCTP by hashing the session connection tag.

The central point architecture distributes GTP-U traffic handled by a gateway GPRS support node (GGSN) and SGSN pair on all SPUs by switching to tunnel endpoint identifier (TEID)-based hash distribution. To handle load-balancing issues, tag-based hash distribution is used to ensure even distribution of SCTP traffic from different associations among all SPUs. (The connection tag for GTP-U is the TEID and for SCTP is the vTag.)

Release History Table

Release	Description
15.1X49-D70	Starting in Junos OS 15.1X49-D70 and Junos OS Release 17.3R1, a new session connection (conn-tag) tag option is available to allow you to add a flow filter to further distinguish GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.
15.1X49-D70	Starting in Junos OS 15.1X49-D70 and Junos OS Release 17.3R1, a new session connection (conn-tag) tag option is available to allow you to add a flow filter to further distinguish GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.
15.1X49-D40	Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, the central point architecture provides enhanced support for GPRS tunneling protocol, control (GTP-C), GPRS tunneling protocol, user plane (GTP-U), and Stream Control Transmission Protocol (SCTP).
15.1X49-D30	Starting with Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, on SRX5000 line devices, the central point architecture is enhanced to handle higher connections per second (cps).

RELATED DOCUMENTATION

Flow-Based Sessions | 68

2

CHAPTER

Flow-Based Sessions

Flow-Based Sessions | **68**

TCP Sessions | **75**

ECMP Flow-Based Forwarding | **86**

Flow-Based Performance | **96**

Flow Distribution and Packet-Ordering | **99**

Fragmentation Packets with PowerMode IPsec | **107**

Unified Policies Support for Flow | **109**

TAP Mode for Flow Sessions | **113**

Flow Management in SRX Series Devices Using VRF Routing Instance | **117**

Flow-Based Sessions

IN THIS SECTION

- Understanding Session Characteristics for SRX Series Services Gateways | 68
- Example: Controlling Session Termination for SRX Series Services Gateways | 69
- Clearing Sessions for SRX Series Services Gateways | 72
- Configuring the Timeout Value for Multicast Flow Sessions | 73

The Junos OS caches the session information that is triggered by the first packet of the flow. The cached session is used by subsequent packets of that same flow and the reverse flow of that session using the flow module, which is integrated into the forwarding path.

Understanding Session Characteristics for SRX Series Services Gateways

Sessions are created, based on routing and other classification information, to store information and allocate resources for a flow. Sessions have characteristics, some of which you can change, such as when they are terminated. For example, you might want to ensure that a session table is never entirely full to protect against an attacker's attempt to flood the table and thereby prevent legitimate users from starting sessions.

Depending on the protocol and service, a session is programmed with a timeout value. For example, the default timeout for TCP is 1800 seconds. The default timeout for UDP is 60 seconds. When a session is terminated, it is marked as invalid, and its timeout is reduced from 20 to 4 seconds.

If no traffic uses the session before the service timeout, the session is aged out and freed to a common resource pool for reuse. You can affect the life of a session in the following ways:

- You can specify circumstances for terminating sessions by using any of the following methods:
  - Age out sessions based on how full the session table is
  - Set an explicit timeout for aging out TCP sessions
  - Configure a TCP session to be invalidated when it receives a TCP RST (reset) message
  - Configure the **fin-invalidate-session** statement to terminate sessions when either session endpoint sends a FIN(ish) message to its peer.

When the peer endpoint receives the packet with the FIN flag set, it sends an ACK(nowlege) message. Typically, tearing down a session using this method involves transmission of a pair of FIN-ACK messages from each session.

- You can configure sessions to accommodate other systems as follows:
  - Disable TCP packet security checks
  - Change the maximum segment size

Understanding Aggressive Session Aging

The session table is a limited resource for SRX Series devices. If the session table is full, any new sessions will be rejected by the device.

The aggressive session-aging mechanism accelerates the session timeout process when the number of sessions in the session table exceeds the specified high-watermark threshold. This mechanism minimizes the likelihood that the SRX Series devices will reject new sessions when the session table becomes full.

Configure the following parameters to perform aggressive session aging:

- *high-watermark*–The device performs aggressive session aging when the number of sessions in the session table exceeds the *high-watermark* threshold.
- *low-watermark*–The device exits aggressive session aging and returns to normal when the number of sessions in the session table dips below the *low-watermark* threshold.
- *early-ageout* –During aggressive session aging, the sessions with an age-out time lower than the *early-ageout* threshold are marked as invalid.

On SRX1400, SRX3400, SRX3600, SRX5600, and SRX5800 devices, the SPU checks the session table, locates the sessions for which the timeout value is lower than the early-ageout time value, and then marks them as invalid. (Platform support depends on the Junos OS release in your installation.)

Example: Controlling Session Termination for SRX Series Services Gateways

IN THIS SECTION

- Requirements | **70**
- Overview | **70**
- Configuration | **70**
- Verification | **71**

This example shows how to terminate sessions for SRX Series devices based on aging out after a certain period of time, or when the number of sessions in the session table is full or reaches a specified percentage. You specify a timeout value or the number of sessions in the session table.

Requirements

Before you begin, understand the circumstances for terminating sessions.

Overview

You can control session termination in certain situations—for example, after receiving a TCP FIN Close or receiving an RST message, when encountering ICMP errors for UDP, and when no matching traffic is received before the service timeout. When sessions are terminated, their resources are freed up for use by other sessions.

In this example, you configure the following circumstances to terminate the session:

- A timeout value of 20 seconds.

  > **NOTE:** The minimum value you can configure for TCP session initialization is 4 seconds. The default value is 20 seconds; if required you can set the TCP session initialization value to less than 20 seconds.

- An explicit timeout value of 280 seconds, which changes the TCP session timeout during the three-way handshake.

  The command sets the initial TCP session timeout to 280 in the session table during the TCP three-way handshake. The timer is initiated when the first SYN packet is received, and reset with each packet during the three-way handshake. Once the three-way handshake is completed, the session timeout is reset to the timeout defined by the specific application. If the timer expires before the three-way handshake is complete, the session is removed from the session table.

- Any session that receives a TCP RST (reset) message is invalidated.

Configuration

Step-by-Step Procedure

To control session termination for SRX Series devices:

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

To control session termination for SRX Series devices:

1. Specify an age-out value for the session.

```
[edit security flow]
user@host# set aging early-ageout 20
```

2. Configure an aging out value.

```
[edit security flow]
user@host# set tcp-session tcp-initial-timeoout 280
```

3. Invalidate any session that receives a TCP RST message.

```
[edit security flow]
user@host# set tcp-session rst-invalidate-session
```

4. If you are done configuring the device, commit the configuration.

```
[edit ]
user@host# commit
```

Verification

To verify the configuration is working properly, enter the **show security flow** command.

Clearing Sessions for SRX Series Services Gateways

IN THIS SECTION

- Terminating Sessions for SRX Series Services Gateways | 72
- Terminating a Specific Session for SRX Series Services Gateways | 72
- Using Filters to Specify the Sessions to Be Terminated for SRX Series Services Gateways | 72

You can use the **clear** command to terminate sessions. You can clear all sessions, including sessions of a particular application type, sessions that use a specific destination port, sessions that use a specific interface or port, sessions that use a certain IP protocol, sessions that match a source prefix, and resource manager sessions.

Terminating Sessions for SRX Series Services Gateways

You can use the following command to terminate all sessions except tunnel and resource manager sessions. The command output shows the number of sessions cleared. Be aware that this command terminates the management session through which the clear command is issued.

```
user@host> clear security flow session all
```

Terminating a Specific Session for SRX Series Services Gateways

You can use the following command to terminate the session whose session ID you specify.

```
user@host> clear security flow session session-identifier 40000381
```

Using Filters to Specify the Sessions to Be Terminated for SRX Series Services Gateways

You can terminate one or more sessions based on the filter parameter you specify for the **clear** command. The following example uses the protocol as a filter.

```
user@host> clear security flow session protocol 89
```

Configuring the Timeout Value for Multicast Flow Sessions

You can configure the timeout value for multicast flow sessions by configuring a custom application and associating the application with a policy.

Multicast flow sessions have one template session and one or more leaf sessions. Because these sessions are linked together, they can have only one timeout value. The timeout value for multicast flow sessions is determined by considering the timeout values configured in the leaf session policies and the IP protocol timeout values. The highest of these timeout values is selected as the multicast flow session timeout.

If no leaf session timeout values are configured, the IP protocol timeout value is automatically used as the timeout value for the mulicast flow session. The IP protocol timeout is the default and is not configurable.

Configuring leaf session timeouts can be especially helpful for multicast streams that have a longer packet interval than the default IP protocol timeout. For example, multicast streams with a packet interval of more than 60 seconds would experience premature aging-out of flow sessions and packet drops with the UDP timeout value, which is always 60 seconds. For such streams, you can configure a higher leaf session timeout value and prevent packet drop.

To set the leaf session timeout value, configure a custom application and associate the application with a policy:

1. Create a custom application, specify its properties, and specify bypassing the application type.

```
[edit]
user@host# edit applications application my-udp
```

```
[edit applications application my-udp]
user@host# set protocol udp
user@host# set destination-port 5000
user@host# set application-protocol ignore
```

2. Set the timeout value for the application protocol.

```
[edit applications application my-udp]
user@host# set inactivity-timeout 500
```

3. Create a policy.

```
[edit]
user@host# edit security policies from-zone vr-zone-1 to-zone junos-host policy my-policy
```

```
[edit security policies from-zone vr-zone-1 to-zone junos-host policy my-policy]
user@host# set match source-address 192.0.2.1
user@host# set match destination-address any
```

4. Associate the custom application (with the configured timeout) to the policy.

```
[edit security policies from-zone vr-zone-1 to-zone junos-host policy my-policy]
user@host# set match application my-udp
user@host# set then permit
```

5. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

6. To verify the updated session timeout value, enter the **show security flow session** command.

```
user@host> show security flow session destination-prefix 203.0.113.0
```

```
Session ID: 2363, Policy name: N/A, Timeout: 498, Valid
  In: 192.0.2.1/17767-->203.0.113.0/5000;udp, If: ge-0/0/1.0, Pkts:0, Bytes:0
  Out: 203.0.113.0/5000-->192.0.2.1/17767/17767;udp, If:.local..4, Pkts:0,
Bytes:0
```

```
Session ID: 2364, Policy name: my-policy/4, Timeout: -1, Valid
  In: 192.0.2.1/17767-->203.0.113.0/5000;udp, If:ge-0/0/1.0, Pkts:1011,
Bytes:258816
  Out: 203.0.113.0/5000-->192.0.2.1/17767;udp, If:ppe0.32769, Pkts:0, Bytes:0
Total sessions: 2
```

In this output, the session ID 2363 section displays a template session. A timeout value of 498 indicates that the template session timeout value is ticking down from the configured value of 500 seconds.

The session ID 2364 section displays a leaf session. The timeout value of -1 essentially indicates that the session will not age out unless the template session ages out.

In this example, the configured leaf session timeout value of 500 seconds is the highest timeout value and is accepted as the template session timeout value for the multicast flow session.

RELATED DOCUMENTATION

ECMP Flow-Based Forwarding | 86
Flow-Based Performance | 96
Flow Distribution and Packet-Ordering | 99

TCP Sessions

IN THIS SECTION

- Understanding TCP Session Checks per Policy | 75
- Example: Configuring TCP Packet Security Checks Per Policy | 76
- Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways | 78
- Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways | 79
- TCP Out-of-State Packet Drop Logging Overview | 81
- Understanding How Preserving Incoming Fragmentation Characteristics Can Improve Throughput | 84

To send data over TCP in a network, a three-way handshake session establishment process is followed. There is a process to start a session, and there is also a process to terminate the TCP session. This topic helps you to understand the process involved in processing a TCP session.

Understanding TCP Session Checks per Policy

By default, the TCP SYN check and sequence check options are enabled on all TCP sessions. The Junos operating system (Junos OS) performs the following operations during TCP sessions:

- Checks for SYN flags in the first packet of a session and rejects any TCP segments with non- SYN flags that attempt to initiate a session.
- Validates the TCP sequence numbers during stateful inspection.

The TCP session check per-policy feature enables you to configure SYN and sequence checks for each policy. Currently, the TCP options flags, no-sequence-check and no-syn-check, are available at a global level to control the behavior of services gateways. To support per-policy TCP options, the following two options are available:

- sequence-check-required: The sequence-check-required value overrides the global value no-sequence-check.
- syn-check-required: The syn-check-required value overrides the global value no-syn-check.

To configure per-policy TCP options, you must turn off the respective global options; otherwise, the commit check will fail. If global TCP options are disabled and SYN flood protection permits the first packet, then the per-policy TCP options will control whether SYN and/or sequence checks are performed.

NOTE:

- The per-policy **syn-check-required** option will not override the behavior of the **set security flow tcp-session no-syn-check-in-tunnel** CLI command.
- Disabling the global SYN check reduces the effectiveness of the device In defending against packet flooding.

CAUTION: Disabling the global SYN check and enforcing the SYN check after policy search will greatly impact the number of packets that the router can process. This in turn will result in intense CPU operations. When you disable global SYN check and enable per-policy SYN check enforcement, you should be aware of this performance impact.

Disabling TCP Packet Security Checks

On an SRX Series device, you can disable security checks on TCP packets to ensure interoperability with hosts and devices with faulty TCP implementations.

The **no-sequence-check** option disables TCP sequence checks. It also increases the throughput.

The **set security flow tcp-session no-sequence-check** command disables the TCP sequence checks on all TCP sessions in default or hash-based modes.

Example: Configuring TCP Packet Security Checks Per Policy

IN THIS SECTION

- Requirements | 77
- Overview | 77

Configuration | 77
Verification | 78

This example shows how to configure TCP packet security checks for each policy in the device.

Requirements

Before you begin, you must disable the tcp options, **tcp-syn-check**, and **tcp-sequence-check** that are configured at global level. .

Overview

The SYN and sequence check options are enabled by default on all TCP sessions. In environments that need to support large file transfers, or that run nonstandard applications, it might be necessary to configure sequence and sync checks differently for each policy. In this example, you configure sequence and sync check for policy **pol1**.

Configuration

Step-by-Step Procedure

To configure TCP packet security checks at the policy level:

1. Configure the checking for the TCP SYN bit before creating a session.

```
[edit]
user@host# set security policies from-zone Zone-A to-zone Zone-B policy pol1 then permit tcp-options
    syn-check-required
```

2. Configure the checking for sequence numbers in TCP segments during stateful inspection.

```
[edit]
user@host# set security policies from-zone Zone-A to-zone Zone-B policy pol1 then permit tcp-options
    sequence-check-required
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
```

```
user@host# commit
```

Verification

To verify that the configuration is working properly, enter the **show security policies detail** command.

Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways

IN THIS SECTION

- Requirements | 78
- Overview | 78
- Configuration | 79
- Verification | 79

This example shows how to disable TCP packet security checks in the device.

Requirements

Before you begin, understand the circumstances for disabling TCP packet security checks. .

Overview

Junos OS provides a mechanism for disabling security checks on TCP packets to ensure interoperability with hosts and devices with faulty TCP implementations. During no-SYN-check the Junos OS does not look for the TCP SYN packet for session creation. No-sequence check disables TCP sequence checking validation. Also, increases throughput. SYN check and sequence check are enabled by default. The set security flow command disables TCP SYN checks and TCP sequence checks on all TCP sessions thus reduces security. This may be required in scenarios with customers like big transfer files, or with applications that do not correctly work with standards.

Configuration

Step-by-Step Procedure

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

To disable TCP packet security checks:

1. Disable the checking of the TCP SYN bit before creating a session.

```
[edit security flow]
user@host# set tcp-session no-syn-check
```

2. Disable the checking of sequence numbers in TCP segments during stateful inspection.

```
[edit security flow]
user@host# set tcp-session no-sequence-check
```

3. If you are done configuring the device, commit the configuration.

```
[edit ]
user@host# commit
```

Verification

To verify the configuration is working properly, enter the **show security flow** command.

Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways

IN THIS SECTION

- Requirements | 80
- Overview | 80
- Configuration | 80
- Verification | 81

This example shows how to set the maximum segment size for all TCP sessions for SRX Series devices.

Requirements

Before you begin, understand the circumstances for setting the maximum segment size.

Overview

You can terminate all TCP sessions by changing the TCP maximum segment size (TCP-MSS). To diminish the likelihood of fragmentation and to protect against packet loss, you can use the tcp-mss to specify a lower TCP MSS value. This applies to all TCP SYN packets traversing the router's ingress interfaces whose MSS value is higher than the one you specify.

If the DF bit is set, it will not fragment the packet and Junos OS will send ICMP error type 3 code 4 packet to the application server (Destination Unreachable; Fragmentation Needed and DF set). This ICMP error message contains the correct MTU (as defined in tcp-mss) to be used by the application server, which should receive this message and adjust the packet size accordingly. This is specifically required with VPNs, as IPsec has added packet overhead; thus tcp-mss must be lowered appropriately.

> **NOTE:** When running SRX Series devices in packet mode, you use the **set system internet-options tcp-mss** to adjust the TCP-MSS value. All ports are affected by the TCP-MSS configuration; you cannot exclude a particular port. When running SRX Series devices in flow mode, although you can use the **set system internet-options tcp-mss** , we recommend using only the **set security flow tcp-mss** to adjust the TCP-MSS value. If both statements are configured, the lower of the two values will take effect.

Configuration

Step-by-Step Procedure

To configure the maximum segment size for all TCP sessions:

1. Set the TCP maximum segment size for all TCP sessions.

```
[edit security flow]
user@host# set tcp-mss all-tcp mss 1300
```

2. If you are done configuring the device, commit the configuration.

```
[edit ]
```

```
user@host# commit
```

Results

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

For brevity, this **show** command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
[edit]
user@host# show security flow
...
tcp-mss{
    all-tcp{
        mss 1300;
    }
}
...
```

Verification

To verify the configuration is working properly, enter the **show configuration security flow** command from operational mode.

```
user@host> show configuration security flow
tcp-mss{
    all-tcp{
        mss 1300;
    }
}
```

TCP Out-of-State Packet Drop Logging Overview

Within any packet-switched network, when demand exceeds available capacity, the packets are queued up to hold the excess packets until the queue fills, and then the packets are dropped. When TCP operates across such a network, it takes any corrective actions to maintain error-free end-to-end communications.

Flow modules already support generating RTLOG for session-based events like session creation and session close. SRX Series devices now support the generation of RTLOG for packet-based events like packet drop without a session existing.

SRX Series devices support logging of unsynchronized TCP out-of-state packets that are dropped by the flow module.

The TCP out-of-state packet drop logging feature avoids any packet loss and enables packet recovery by logging the out-of-sync packets for error free communication, and prevents the database servers from going out of sync. This feature is built on top of the security log (RTLOG) facility.

TCP out-of-state packet drop logging supports capturing of TCP packet drop logs under the following conditions:

- **Session ages out**—When there are cloud applications running on top of long TCP sessions, and when these applications do not refresh the TCP sessions after the session ages out, the TCP packets are dropped. This feature supports logging of these dropped TCP packets.
- **Unsynchronized first packets due to attacks or asymmetric routes**—When you deploy SRX Series devices at two sites , and when routing sometimes forces asymmetric traffic, the synchronization (SYN) packet is seen at one site but the synchronization acknowledgment (SYN_ACK) packets are seen at another site.

  This means that the SRX Series device sees a TCP ACK packet for which it does not have a matching state table entry. This might occur because the connection was inactive for a period of time or the connections tables were flushed (for example, because of a policy installation or restart).

  The SYN_ACK packets that are seen at another site in this case were denied by the SRX Series device but were not logged. This feature supports logging of the denied SYN_ACK packets.
- **Other out-of-state conditions (like TCP sequence check fail and synchronization packet received in FIN state)**—When an SRX Series device detects a sequence failure, if the device is in TCP four-way close state but receives SYN packets, or if there is a three-way handshake failure, the SRX Series device drops the TCP packets and these dropped packets are logged.

> **NOTE:** The unsynchronized TCP out-of-state packet drop log is a packet-based log, not a session-based log.
>
> TCP out-of-state packet drop logging is designed with a throttle mechanism to protect CPU from being attacked, and within each throttle interval some logs can be dropped.
>
> Only TCP out-of-state packets dropped by Flow module are logged. TCP packets dropped by TCP-proxy and IDP are not logged.

Understanding TCP Out-of-State Packet Drop Logging

To understand the implementation of TCP out-of-state packet drop logging, consider that you deploy SRX Series devices at two sites and that routing sometimes forces asymmetrical traffic, where the SYN packet is seen at one site but the SYN_ACK packet is seen at another site. The SYN_ACK packet in this case would be denied but not logged. The TCP out-of-state packet drop logging feature provides visibility into these unsynchronized packet drops.

Consider the scenario where databases within the data center keep their TCP sockets open, with no keepalives being sent. If no data is being transmitted, the SRX Series device will timeout the sessions. Although the databases will send some data through that TCP socket, when the traffic reaches the SRX Series device, the session is no longer there and the packet is dropped, but not logged. These out-of-state TCP packets that are dropped are now logged by the SRX Series device.

Supported TCP Out-of-State Logging Features

TCP out-of-state logging supports the following features:

- A packet filter component to filter target traffic.
- A throttle component to protect CPU from being overloaded by log messages.
- Flexibility to change the log generation rate.

Packet Filter Component

The logging filter leverages the current flow trace filter. It provides different ways to filter traffic. You must configure the filters to generate packet logs, otherwise logs will not be triggered.

This filter functionality avoids enabling logs unexpectedly. The maximum filters supported are 64.

Use the **set security flow packet-log packet-filter *<filter-name>*** command to enable the related filter components you want.

Throttle Component

Logging every TCP out-of-state packet can overload the device when traffic is heavy or when an attack occurs. if the CPU is idle and you want to log as many messages as possible, then this could lead to CPU overload.

The throttle mechanism allows you to configure the throttle interval from the CLI, so you can protect your CPU from being overloaded.

A hash table is introduced to map your logged data. The hash key is generated with the source-IP address, destination-IP address, source port, and destination port.

Within each throttle interval, only a limited number (more than one) of messages will be sent to RTLOG. The remaining log messages will be throttled.

The default throttle interval is 1 second. The throttle interval (at the millisecond level) needs to be configured as a power of two or zero (0, 1, 2, 4, 8, 16 … 2^N).

When the throttle interval is configured as 0, no throttle mechanism will be involved. This is suitable for scenarios where traffic is very light and you want to record all the packet drop logs.

Configuration of the throttle interval as 2^N makes the throttle mechanism lockless and provides good log capture performance.

Flexibility for Changing the Log Generation Rate

Based on the throttle interval set, the log generation rate can be modified and managed.

This means that within each 32-millisecond (ms) interval, a limited number of logs could be generated and the remaining could be dropped. We recommend that you configure the interval as (0, 1, 2, 4, 8, 16, 32 … 2^N).

If the input value is not aligned to 2^N, it will be aligned to 2^N automatically during flow processing. For example, if you configure a 10-ms interval it will be aligned to an 8-ms interval automatically.

Understanding How Preserving Incoming Fragmentation Characteristics Can Improve Throughput

This topic covers the benefits of using the SRX Series device to preserve the characteristics of incoming packet fragments.

When data is sent from one host to another, it is transmitted as a series of packets. Performance is improved and network resources are conserved when packets of the largest size can transit the path from the source node to the destination node without being fragmented at any link in the datapath. When a packet must be fragmented into smaller packets to transit a link in the path because the packet is larger than that of the maximum transmission unit (MTU) established for that link, each of the resulting fragments must contain packet header information, in addition to the payload, or data. The increased overhead can lower throughput and degrade network performance. Also, the packet fragments must be reassembled at the destination node, which consumes additional network resources.

On the other hand, network resources are wasted when a host sends packets that are much smaller than the path MTU (path maximum transmission unit), resulting in suboptimal throughput. The path MTU discovery process works to discover the optimal MTU size for fragments that transit the datapath from the source node to the destination node for a session. The optimal packet size, then, is that of the path MTU. Fragmentation occurs when the size of a packet exceeds the path MTU.

If application-layer services are configured on the SRX Series device, packet fragments at the ingress interface must be reassembled before the services can be applied and the content inspected. These reassembled packet fragments must be broken down again before the data is transmitted out the egress

interface. Normally, it is the MTU size of the egress interface that determines the size of fragments transmitted out the SRX Series device to the next link. It could be the case that the egress MTU size on the SRX Series device is larger than the path MTU, which, again, would result in packet fragmentation in the datapath, reducing performance or causing packet drop. Packet fragments must be small enough to transit every link in the path from source to destination.

By default, the SRX Series device uses the MTU size configured for the egress interface to determine the size for packet fragments it transmits. However, if you enable the feature for preserving incoming fragment characteristics, the SRX Series device detects and saves the size of incoming packet fragments.

To diminish the likelihood of packet fragmentation in the datapath, the SRX Series device keep track of and adjust the egress MTU for that flow. It identifies the maximum size of all incoming fragments. It uses that information in conjunction with the existing MTU of the egress interface to determine the correct MTU size for fragmented packets sent out the egress interface. The SRX Series device compares the two numbers. It takes the smaller number and uses it for the egress interface MTU size.

Configure the device using the **set security flow preserve-incoming-frag-size** command to enable the feature that takes into account the size of incoming packet fragments.

Table 7 on page 85 summarizes how the SRX Series egress MTU size is determined.

Table 7: How the Final Egress MTU Size for Fragments Exiting the SRX Series Device Is Determined

Incoming Fragment Size	Existing Egress MTU Size	Final Egress MTU Size
If the largest fragment is	*smaller* than the existing egress MTU size	largest incoming fragment size is used.
If the largest fragment is	larger than the existing egress MTU size	existing egress interface MTU is used.

NOTE: This feature is supported on SRX Series devices. It supports through-traffic and traffic exiting a tunnel. It is applies to both IPv4 and IPv6 traffic.

The following two considerations affect fragment size:

- For stream-based applications, such as UTM and ALG, the applications themselves could change or reassemble packets even if there were no fragments received. In this case, the existing egress interface MTU is used.
- When a path MTU discovery packet is delivered to a session, the path MTU for that session is reset to the value established by the path MTU packet.

Release History Table

Release	Description
15.1X49-D100	Configure the device using the **set security flow preserve-incoming-frag-size** command to enable the feature that takes into account the size of incoming packet fragments.

RELATED DOCUMENTATION

Flow-Based Performance | **96**

Flow Distribution and Packet-Ordering | **99**

ECMP Flow-Based Forwarding

IN THIS SECTION

Understanding ECMP Flow-Based Forwarding | **86**

Example: Configuring ECMP Flow-Based Forwarding | **89**

This topic provides a brief overview of equal-cost multipath (ECMP) for forwarding and reverse side traffic on Junos OS SRX Series devices and vSRX instances. For comprehensive coverage of the ECMP implementation on Junos OS SRX Series devices and vSRX instances.

Understanding ECMP Flow-Based Forwarding

Equal-cost multipath (ECMP) is a network routing strategy that allows for traffic of the same session, or flow—that is, traffic with the same source and destination—to be transmitted across multiple paths of equal cost. It is a mechanism that allows you to load balance traffic and increase bandwidth by fully utilizing otherwise unused bandwidth on links to the same destination.

When forwarding a packet, the routing technology must decide which next-hop path to use. In making a determination, the device takes into account the packet header fields that identify a flow. When ECMP is used, next-hop paths of equal cost are identified based on routing metric calculations and hash algorithms.

That is, routes of equal cost have the same preference and metric values, and the same cost to the network. The ECMP process identifies a set of routers, each of which is a legitimate equal cost next hop towards the destination. The routes that are identified are referred to as an ECMP set. Because it addresses only the next hop destination, ECMP can be used with most routing protocols.

An equal-cost multipath (ECMP) set is formed when the routing table contains multiple next-hop addresses for the same destination with equal cost. (Routes of equal cost have the same preference and metric values.) If there is an ECMP set for the active route, Junos OS uses a hash algorithm to choose *one* of the next-hop addresses in the ECMP set to install in the forwarding table.

You can configure Junos OS so that multiple next-hop entries in an ECMP set are installed in the forwarding table. On Juniper Networks devices, per-flow load balancing can be performed to spread traffic across multiple paths between routing devices. On Juniper Networks security devices, source and destination IP addresses and protocols are examined to determine individual traffic flows. Packets for the same flow are forwarded on the same interface; the interface does not change when there are additions or changes to the ECMP set. This is important for features such as source NAT, where the translation is performed only during the first path of session establishment for IDP, ALG, and route-based VPN tunnels. If a packet arrives on a given interface in an ECMP set, the security device ensures that reverse traffic is forwarded through the same interface.

> **NOTE:** ECMP flow-based forwarding on security devices applies to IPv4 and IPv6 unicast traffic flows. Starting with Junos OS Release 15.1X49-D60, ECMP flow-based forwarding of IPv6 unicast traffic is supported on all SRX Series devices and vSRX instances. Multicast flow is not supported.

On Juniper Networks security devices, the maximum number of next-hop addresses in an ECMP set that can be installed in the forwarding table is 16. If there are more than 16 next-hop addresses in an ECMP set, only the first 16 addresses are used.

In a chassis cluster deployment, a *local* interface is an interface that is on the same node as the interface on which a packet arrives, and a *remote* interface is an interface that is on the other chassis cluster node. If an ECMP route has both local and remote interfaces in a chassis cluster, then the local interface is favored for the next hop.

If a next-hop address is no longer part of the ECMP set or if it is removed from the routing table because of a route change, a flow that uses the next hop is rerouted and the session is not affected. Rerouting of the flow also occurs if there is a configuration change that takes away the next-hop address or if an administrator takes down the next-hop interface without deleting it. If a next-hop address is removed from the routing table because the interface is deleted or the session is intentionally cleared, the session is killed without being rerouted.

> **NOTE:** We recommend that interfaces in an ECMP set be in the same security zone. If a flow is rerouted and the rerouted flow uses an interface in a different security zone than the original route, the session is killed.

To configure ECMP flow-based forwarding on Juniper Networks security devices, first define a load-balancing routing policy by including one or more **policy-statement** configuration statements at the [**edit policy-options**] hierarchy level, with the action **load-balance per-packet**. Then apply the routing policy to routes exported from the routing table to the forwarding table. To do this, include the **forwarding-table** and **export** configuration statements at the [**edit routing-options**] hierarchy level.

ECMP Implementation for Junos OS SRX Series Devices and vSRX Instances

You can configure ECMP for SRX Series devices and vSRX instances to implement per-flow load balancing to spread traffic across multiple paths between routing devices. Routes of equal cost have the same preference and metric values. These devices examine the source IP address, the destination IP address, and the protocol to determine individual traffic flows. Traffic with the same source IP address, destination IP address, and protocol number that is permitted by a security policy is forwarded to the same next hop. Junos OS on these devices uses the flow information in its hashing logic.

For Junos OS SRX Series devices and vSRX instances, an ECMP set is formed when the routing table contains multiple next-hop addresses for the same destination with equal cost. ECMP allows for multiple next-hop entries in an ECMP set to be installed in the forwarding table. Packets for the same flow are forwarded on the same interface; the interface does not change when there are additions or changes to the ECMP set.

If there is an ECMP set for the active route, Junos OS uses a hash algorithm to choose *one* of the next-hop addresses in the ECMP set to install in the forwarding table.

> **NOTE:** ECMP flow-based forwarding on SRX Series devices and vSRX instances applies to IPv4 and IPv6 unicast traffic flows. Starting in Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, ECMP flow-based forwarding of IPv6 unicast traffic is supported on all SRX Series devices and vSRX instances. Multicast flow is not supported.

ECMP for Reverse Traffic

Starting in Junos OS Release 17.3, if you enable ECMP support for reverse traffic, the SRX Series device uses a hash algorithm to determine the interface to use for reverse traffic in a flow. This process is similar to asymmetric routing in which a packet traverses from a source to a destination in one path and takes a different path when it returns to the source.

If you do not enable this feature, the SRX Series device selects a route in the ECMP set to the incoming interface for reverse traffic, which is the default behavior.

You use the **allow-reverse-ecmp** configuration statement in the [**edit security flow**] hierarchy to configure ECMP flow-based forwarding to use a hash algorithm in selecting a route in the ECMP set for reverse traffic transit. That is, if you enable this function, rather than selecting a route to the incoming interface, the SRX Series device uses a hash algorithm to select a route in the ECMP set for reverse traffic.

Because the ECMP flow-based policy is zone-based, ECMP reverse lookup support ensures that the egress interface used for reverse traffic is in the same zone as the ingress interface used for arriving traffic.

NOTE: Interfaces in an ECMP set must be in the same security zone. If the egress interface zone is different from the ingress interface zone, a session can be created but the packets will be dropped.

CAUTION: If you decide to enable reverse ECMP, be aware of the following condition and take action to avoid it: When ECMP flow-based forwarding is used, the SRX Series device could cause upstream devices to see only one-way traffic of a session. Problems might ensue for upstream devices that maintain session state, for example, for TCP-proxy and SYN-proxy. The issue is similar to asynchronous routing behavior.

Example: Configuring ECMP Flow-Based Forwarding

IN THIS SECTION

- Requirements | **90**
- Overview | **90**
- Configuration | **91**
- Verification | **95**

This example shows how to configure ECMP flow-based forwarding.

Requirements

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

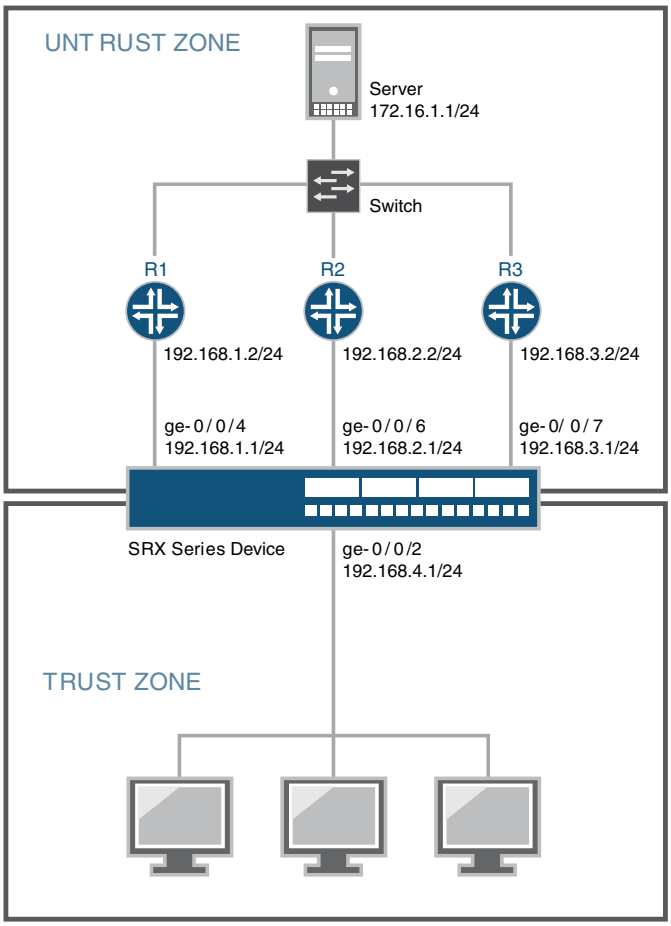
Overview

This example configures three static ECMP routes on an SRX Series device. Each static route uses a different next-hop router to reach the destination server. The interfaces towards the routers are assigned to the untrust security zone. This example creates a load-balancing routing policy named **load-balancing-policy** and applies the policy to all routes exported from the routing table to the forwarding table.

Topology

Figure 10 on page 90 shows the topology used in this example.

Figure 10: ECMP Routes



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.

```
## Interfaces ##
set interfaces ge-0/0/2 unit 0 family inet address 192.168.4.1/24
set interfaces ge-0/0/4 unit 0 family inet address 192.168.1.1/24
set interfaces ge-0/0/6 unit 0 family inet address 192.168.2.1/24
set interfaces ge-0/0/7 unit 0 family inet address 192.168.3.1/24
## Static routes ##
set routing-options static route 172.16.1.0/24 next-hop 192.168.1.2
set routing-options static route 172.16.1.0/24 next-hop 192.168.2.2
set routing-options static route 172.16.1.0/24 next-hop 192.168.3.2
## Security zones, address book entry, and policy ##
set security zones security-zone trust interfaces ge-0/0/2
set security zones security-zone untrust interfaces ge-0/0/4
set security zones security-zone untrust interfaces ge-0/0/6
set security zones security-zone untrust interfaces ge-0/0/7
set security address-book global address FTP-servers 172.16.1.0/24
set security policies from-zone trust to-zone untrust policy permit-ftp match source-address any
set security policies from-zone trust to-zone untrust policy permit-ftp match destination-address FTP-servers
set security policies from-zone trust to-zone untrust policy permit-ftp match application junos-ftp
set security policies from-zone trust to-zone untrust policy permit-ftp then permit
## ECMP routing policy ##
set policy-options policy-statement load-balancing-policy then load-balance per-packet
set routing-options forwarding-table export load-balancing-policy
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy.

To configure ECMP flow-based forwarding:

1. Configure interfaces.

```
[edit interaces]
user@host# set ge-0/0/2 unit 0 family inet address 192.168.4.1/24
user@host# set ge-0/0/4 unit 0 family inet address 192.168.1.1/24
user@host# set ge-0/0/6 unit 0 family inet address 192.168.2.1/24
user@host# set ge-0/0/7 unit 0 family inet address 192.168.3.1/24
```

2. Configure static routes.

```
[edit routing-options]
user@host# set static route 172.16.1.0/24 next-hop 192.168.1.2
user@host# set static route 172.16.1.0/24 next-hop 192.168.2.2
user@host# set static route 172.16.1.0/24 next-hop 192.168.3.2
```

3. Create the **trust** and **untrust** security zones, and include the related interfaces.

```
[edit security]
user@host# set zones security-zone trust interfaces ge-0/0/2
user@host# set zones security-zone untrust interfaces ge-0/0/4
user@host# set zones security-zone untrust interfaces ge-0/0/6
user@host# set zones security-zone untrust interfaces ge-0/0/7
```

4. Configure an address book entry for the server subnet.

   This entry is used in the security policy.

```
[edit security address-book]
user@host# set global address FTP-servers 172.16.1.0/24
```

5. Configure a security policy.

```
[edit security policies from-zone trust to-zone untrust]
user@host# set policy permit-ftp match source-address any
user@host# set policy permit-ftp match destination-address FTP-servers
user@host# set policy permit-ftp match application junos-ftp
user@host# set policy permit-ftp then permit
```

6. Create a load-balancing routing policy.

```
[edit policy-options]
user@host# set policy-statement load-balancing-policy then load-balance per-packet
```

7. Apply the routing policy to all routes exported from the routing table to the forwarding table.

```
[edit routing-options]
user@host# set forwarding-table export load-balancing-policy
```

Results

From configuration mode, confirm your configuration by issuing the **show interfaces**, **show security**, **show policy-options**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@host# show interfaces
ge-0/0/2 {
    unit 0 {
        family inet {
            address 192.168.4.1/24;
        }
    }
}
ge-0/0/4 {
    unit 0 {
        family inet {
            address 192.168.1.1/24;
        }
    }
}
ge-0/0/6 {
    unit 0 {
        family inet {
            address 192.168.2.1/24;
        }
    }
}
ge-0/0/7 {
    unit 0 {
        family inet {
            address 192.168.3.1/24;
        }
    }
}
user@host# show security
address-book {
    global {
        address FTP-servers 172.16.1.0/24;
    }
}
policies {
    from-zone trust to-zone untrust {
        policy permit-ftp {
            match {
                source-address any;
```

```
            destination-address FTP-servers;
            application junos-ftp;
        }
        then {
            permit;
        }
    }
  }
}
zones {
  security-zone trust {
    interfaces {
      ge-0/0/2.0;
    }
  }
  security-zone untrust {
    interfaces {
      ge-0/0/4.0;
      ge-0/0/6.0;
      ge-0/0/7.0;
    }
  }
}
user@host# show policy-options
policy-statement load-balancing-policy {
  then {
    load-balance per-packet;
  }
}
```

```
[edit]
user@host# show routing-options
static {
  route 172.16.1.0/24 next-hop [ 192.168.1.2 192.168.2.2 192.168.3.2 ];
}
forwarding-table {
  export load-balancing-policy;
}
```

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

Verification

Verifying the Forwarding Table

Purpose

Verify that the route information for all ECMP routes appears in the forwarding table.

Action

From operational mode, enter the **show route forwarding-table destination 172.16.1.0** command.

```
user@host> show route forwarding-table destination 172.16.1.0
Routing table: default.inet
Internet:
Destination        Type RtRef Next hop           Type Index NhRef Netif
172.16.1.0/24      user     0                    ulst 262142     2
                            192.168.1.2          ucst   560     2 ge-0/0/4.0
                            192.168.2.2          ucst   561     2 ge-0/0/6.0
                            192.168.3.2          ucst   562     2 ge-0/0/7.0
...
```

Meaning

The output shows a next hop type of **ulst**, which means the route has multiple eligible next hops. Packets destined for the 172.16.1.0 network can use any next hop in the list.

Release History Table

Release	Description
15.1X49-D60	Starting with Junos OS Release 15.1X49-D60, ECMP flow-based forwarding of IPv6 unicast traffic is supported on all SRX Series devices and vSRX instances. Multicast flow is not supported.
15.1X49-D60	Starting in Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, ECMP flow-based forwarding of IPv6 unicast traffic is supported on all SRX Series devices and vSRX instances. Multicast flow is not supported.

RELATED DOCUMENTATION

Flow-Based Sessions | 68

Flow Distribution and Packet-Ordering | 99

Flow-Based Performance

IN THIS SECTION

- Expanding Session Capacity by Device | 96
- Verifying the Current Session Capacity | 97

This topics explains about the performance of the session capacity. Expanding the session capacity and reverting back to the default session capacity.

Expanding Session Capacity by Device

To take advantage of the processing potential of a fully loaded SRX5600, SRX5800 device, or vSRX, you can expand the maximum number of concurrent sessions for these devices.

Table 8 on page 96 shows the maximum number of concurrent sessions allowed on these devices by default and with expanded capacity. Platform support depends on the Junos OS release in your installation.

Table 8: Maximum Central Point Session Increases

	Maximum Concurrent Sessions on a Fully Loaded System	
SRX Series Devices	Default	With Expanded Capacity
SRX3400	2.25 million	3 million
SRX3600	2.25 million	6 million
SRX5400	42 million	Expansion not available
SRX5600	114 million	Expansion not available
SRX5800	258 million	Expansion not available

The method used for expanding session capacity depends on the device:

- Central point session license installation and validation on an SRX3400 or SRX3600 device

- CLI optimization option on an SRX5800 device

Expanding Session Capacity on an SRX3400 or SRX3600 Device

Expanding session capacity on an SRX3400 or SRX3600 device requires validation of a central point session license on the device.

1. Obtain the central point session license key and install the license on the device..
2. Reboot the device to implement the expanded session capacity.

Reverting to Default Session Capacity on an SRX5800 Device

Reverting to the default session capacity on an SRX5800 device requires a CLI configuration change.

1. Enter the following command at the CLI configuration prompt to reestablish the default session capacity value:

   ```
   user@host# set security gprs gtp enable
   ```

2. Commit the configuration.

   ```
   user@host# commit
   ```

3. Reboot the device to implement the new value.

Verifying the Current Session Capacity

Purpose

The central point session summary includes the maximum sessions setting for the device. From this value you can determine if the session capacity has been modified as you expected.

Action

To verify the current setting of the central point session capacity, enter the following CLI command.

```
user@host> show security flow cp-session summary
```

```
DCP Flow Sessions on FPC10 PIC0:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2
Maximum sessions: 7549747
Maximum inet6 sessions: 7549747

DCP Flow Sessions on FPC10 PIC2:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2
Maximum sessions: 7549747
Maximum inet6 sessions: 7549747

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
Maximum sessions: 7549747
Maximum inet6 sessions: 7549747
```

Meaning

The **Maximum sessions** value reflects the current session capacity on your device. A value of 14000000 means that the SRX5800 device is configured for the expanded number of central point sessions.

Flow Distribution and Packet-Ordering

IN THIS SECTION

- Understanding Load Distribution in SRX5000 Line Devices | **99**
- Understanding Packet-Ordering Function on SRX5000 Line Devices | **102**
- Understanding Session Distribution on SRX5000 Line Devices in Adaptive Mode | **104**

This topic describes about the load distribution and the packet ordering on SRX5000 Line devices.

Understanding Load Distribution in SRX5000 Line Devices

The load distribution algorithm, which is supported on the SRX5800, SRX5600, and SRX5400 devices, is adjusted based on session capacity and processing power. (Actual platform support depends on the Junos OS release in your installation.)

Hash-based session distribution uses a hash table. The SPU session weight table is used to assign an SPU ID to each hash index in the session distribution hash table. This way, the number of sessions created on each SPU using hash-based distribution is proportional to the SPU's weight in the SPU session weight table. Each NPU also keeps an identical SPU session weight table and session distribution hash table that it uses to select an SPU to forward packets that do not match an NPU session.

In the event of a SPU failure, the Routing Engine will reset all of the cards on the dataplane including IOCs and NPCs in order to maintain hash table consistency for session distribution.

In hash-based session distribution, weights are based on session capacity. We recommend the hash session distribution mode when high session capacity is required.

NOTE: Load distribution on SRX5000 line devices is always hash-based.

Insertion and removal of SPCs causes recalculation of the SPU session weight table at central point initialization time because the chassis must reboot after insertion.

Starting in Junos OS Release 15.1X49-D30, the central point architecture is enhanced to handle higher concurrent sessions and connections per second (cps) for the SRX5000 Series device.

The central point architecture enhancements prevent data packets from going through the central point by offloading traffic management to SPUs. The system session capacity is extended, as the session limit on the central point is removed.

Calculating SPU ID

The SPU ID for a device equipped with SRX3K-SPC-1-10-40, SRX5K-SPC-2-10-40, or SRX5K-SPC3 Services Processing Card (SPC) is calculated as follows:

```
SPU ID = (FPC ID X 4) + PIC ID
```

The SRX3K-SPC-1-10-40, SRX5K-SPC-2-10-40, and SRX5K-SPC3 contains two PICs per card, four PICs per card (FPC), and two PICs per card respectively. For example, a device contains 2 cards in slot 1 (FPC ID 0) and slot 2 (FPC ID 1), the expected SPU IDs are as follows:

- For SPC1: (0, 1) and (4, 5), total 4 SPUs in 2 cards.
- For SPC2: (0, 1, 2, 3) and (4, 5, 6, 7), total 8 SPUs in 2 cards.
- For SPC3: (0, 1) and (4, 5), total 4 SPUs in 2 cards.

For FPC1 (the second card) and PIC1 (the second PIC in the card), the SPU ID is calculated as:

```
SPU ID = (FPC ID X 4) + PIC ID
       = (1 X 4) + 1
       = 4 + 1
       = 5
```

Use this convention while referring the SPU ID for CLI and SNMP.

Hash-Based Forwarding on the SRX5K-MPC, SRX5K-MPC3-40G10G (IOC3), and the SRX5K-MPC3-100G10G (IOC3)

On these SRX Series devices, a packet goes through a series of events involving different components as it progresses from ingress to egress processing. With the datapath packet forwarding feature, you can obtain quick delivery of I/O traffic over the SRX 5000 line of devices.

The SRX5K-MPC, SRX5K-MPC3-40G10G (IOC3), and SRX5K-MPC3-100G10G (IOC3) are interface cards supported on the SRX5400, SRX5600, and SRX5800 devices. The Modular Port Concentrator (MPC) provides load-balancing services for Services Processing Units (SPUs) by using the hash-based forwarding method.

In hash-based forwarding, the packet might be forwarded by the MPC to a selected SPU (DCP) instead of the central point. This approach enhances session scaling and prevents overloading of the central point.

Hash value calculation involves the following steps:

- For IPv4 packets, the hash-based forwarding module generates the hash value based on Layer 3 and Layer 4 information, depending on different Layer 4 protocol types.
- For Stream Control Transmission Protocol (SCTP), TCP, UDP, Authentication Header (AH), edge service provider (ESP), and Internet Control Message Protocol (ICMP) protocols, the hash module utilizes Layer 4 information to generate the hash value. For any other protocols, only Layer 3 information is used in hash generation.
- For IPv4 fragment packets, the hash value is calculated using only the Layer 3 information. This also applies to the first fragment of the packet.
- For non-IP packets, the hash-based forwarding module uses the Layer 2 information to calculate the hash value.

Once a hash value is calculated according to the packet's Layer 2, Layer 3, or Layer 4 information, an SPU ID is assigned to each hash index in the session distribution hash table.

> **NOTE:** The SRX5K-MPC (IOC2), SRX5K-MPC3-40G10G (IOC3), and SRX5K-MPC3-100G10G (IOC3) can only be used on SRX5400, SRX5600, and SRX5800 devices that are configured for hash-based session distribution.
>
> When the hash-based session distribution mode is enabled, the system changes its behavior to high-session-capacity-based mode when the SRX5K-MPC, SRX5K-MPC3-40G10G (IOC3), and SRX5K-MPC3-100G10G (IOC3) are installed on the device.

> **NOTE:** On SRX5000 line devices with an SRX5K-MPC, SRX5K-MPC3-40G10G (IOC3), or SRX5K-MPC3-100G10G (IOC3) installed, during a system or an SPU reboot, when the hash-based session distribution mode is enabled, traffic will pass only when all SPUs are up after the reboot.

The MPCs on the IOC3 provide load-balancing services for SPUs by performing hash-based datapath packet forwarding to interconnect with all existing IOCs and SPCs.

The IOC3 processes ingress and egress packets. The IOC3 parses the ingress packet and sends it to the SPU for further security processing, including flow session lookup, zone and policy check, VPN, ALG, and so on.

The IOC3 manages packet data memory and fabric queuing for packet lookup and encapsulation functions.

> **NOTE:** Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, hash-based session distribution is the default mode for the SRX5400, SRX5600, and SRX5800 devices. Selection of hash keys depends on application protocols.
>
> Starting with Junos OS Release 17.4R1, traffic is hashed and distributed to different SPUs by the IOC, based on a hash-based session distribution algorithm. This enhancement provides an even hash distribution among all SPUs by using a larger fixed-length hash table. In earlier Junos OS releases, the traffic distribution was uneven among all SPUs due to a fixed-length hash table.

The IOC3 sets up a security flow table (IPv4 and IPv6) including key, result table, and packet memory.

The following functions are provided with the flow table:

- Flow lookup
- Flow insertion and deletion
- Security flow aging out
- Security flow statistics

Understanding Packet-Ordering Function on SRX5000 Line Devices

The packet-ordering function, which is supported on the SRX5400, SRX5600, and SRX5800, devices and vSRX, improves the performance of the device by activating the built-in packet-ordering function of the Packet Ordering Engine on the XLP processor on the application central point.

Two types of the packet ordering modes are supported: hardware and software.

If the packet-ordering function is set to *hardware*, the load-balancing thread (LBT) and the packet-ordering thread (POT) are offloaded to the packet ordering engine and resources are freed to perform packet processing. If the packet-ordering function is set to *software*, the load-balancing thread (LBT) and the packet-ordering thread (POT) are running on the SPU. By default, packet-ordering mode using the Packet Ordering Engine (hardware) is enabled on the device. You can disable it with a configuration change that requires a reboot.

The flow thread receives the packets, processes them, and sends or drops them. For packets that require no ordering, the flow thread notifies the Network Acceleration Engine (NAE) egress to send or drop the packets. For packets that require ordering, the flow thread notifies the Packet Ordering Engine to dequeue the packets from the ordering list and to send or drop the packets in order.

Changing Packet-Ordering Mode on SRX5000 Line Devices

The packet-ordering functionality using the Packet Ordering Engine is supported on SRX5400, SRX5800 and SRX5600 devices with next-generation SPCs. (Platform support depends on the Junos OS release in your installation.) By default, packet-ordering mode using the Packet Ordering Engine is enabled. To disable the packet-ordering functionality using the Packet Ordering Engine, you must update the packet-ordering mode on the device.

The following packet ordering modes are supported:

- software—Disables the packet-ordering mode using the Packet Ordering Engine.
- hardware—Enables the packet-ordering mode using the Packet Ordering Engine. This is the default option.

To disable the packet-ordering mode using the Packet Ordering Engine:

1. Enter the following command at the CLI configuration prompt to specify the packet-ordering mode.

```
[edit]
user@host#  set security forwarding-process application-services packet-ordering-mode software
```

2. Use the **show security forwarding-process** command to review your configuration.

```
[edit]
user@host# show security forwarding-process
application-services{
   packet-ordering-mode software;
}
```

3. Check your changes to the configuration before committing.

```
[edit]
user@host# commit check
```

```
warning: System packet ordering mode changed, reboot is required to take effect.
If you have deployed a cluster, be sure to reboot all nodes.
configuration check succeeds
```

4. Commit the configuration.

```
[edit]
user@host# commit
```

```
warning: System packet ordering mode changed, reboot is required to take effect.
If you have deployed a cluster, be sure to reboot all nodes.
commit complete
```

5. Reboot the device at an appropriate time.

6. Use the **show security flow status** command to verify the packet-ordering mode.

user@host> show security flow status

```
Flow forwarding mode:
  Inet forwarding mode: flow based
  Inet6 forwarding mode: drop
  MPLS forwarding mode: drop
  ISO forwarding mode: drop
Flow trace status
  Flow tracing status: off
Flow session distribution
  Distribution mode: RR-based
Flow packet ordering
  Ordering mode: Software (reboot needed to change to Software)
```

Understanding Session Distribution on SRX5000 Line Devices in Adaptive Mode

Starting in Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, adaptive mode session distribution was replaced by enhancements to the central point architecture.

Adaptive mode session distribution is implemented on the SRX5000 series devices running in mixed mode prior to Junos OS Release 15.1X49-D30 and Junos OS Release 17.1R1. Adaptive mode session distribution maximizes use of system resources by taking into account a Services Processing Unit's (SPU) capacity and its available resources. It is enabled only on SRX5000 Series devices running in XLR/XLP mixed mode, that is in chassis deployments in which different types of SPUs are used in different combinations. If an SRX5800, SRX5600, or SRX5400 device contains a mix of next-generation services processing cards (SPCs) and existing SPCs, then adaptive mode session distribution is assumed as the default. For SRX5000 Series devices not running in mixed mode, hash-based load balancing is the default.

A Services Processing Card (SPC) contains one or more SPUs each of which processes the packets of a flow according to the security features and other services configured for sessions distributed to it by the central point (CP). An SPU's CPU load changes from time to time. To fully utilize changing available capacity and adapt session distribution accordingly, in adaptive mode the system assigns a weight to all SPUs dynamically. It is the weight of the SPUs that determine the session distribution.

Each SPU sends its CPU usage information to the central point (CP) periodically. The central point checks these values, calculates the weight every 1 second, and distributes the sessions in such a way as to maximize overall system performance. In other words, In adaptive mode, session distribution is based on a *dynamic* weighted assignment system that is calculated in real time allowing for full capacity utilization of the CPUs of all SPUs, regardless of their type.

It is the dynamic calculation of weights that distinguishes adaptive mode session distribution from weighted round-robin (WRR) session distribution. While WRR differentiates SPUs and their CPU capacity by calculating and assigning weights to the different types of SPUs, the calculation and assignment is static, that is, it is done only once, at initialization. Adaptive mode improves on the fixed ratio session distribution process of WRR. WRR leads to underutilization of system resources because session processing limits are set based only on the type of SPU and its CPU capacity, not taking into account its available processing power.

For adaptive mode session distribution, the following formula is used to calculate the weight assigned to an SPU:

Wi = Sum(W1-n)*Ci*Si/Sum(C1-n*S1-n)

Where:

- **Wi**— weight assigned to the SPU.
- **Sum(W1-n)**— Total weight of system. This values is constant.
- **n**—total number of SPUs.
- **Ci**—available CPU computational power of the SPU.
- **Si**—available session capacity of SPU.

In adaptive mode, when the CPU usage on one SPU is high, fewer sessions are distributed to that SPU. The following examples explains the calculation.

Consider a device with two SPUs. Each SPU's session capacity is 1 million.

For a certain time:

- When SPU1 has 500,000 sessions on it, CPU usage of it is 10 percent:
  - Available CPU capacity of SPU1 (C1) = 1-10 percent = 90 (percent).
  - Available session capacity of SPU1 (S1) = 1-500,000/1M = 50 (percent).
- When SPU2 has 400,000 sessions on it, CPU usage of it is 20 percent:
  - Available capacity of SPU2 (C2)= 1-20 percent= 80 (percent).
  - Available session capacity of SPU2 (S2)= 1-400,000/1M= 60 (percent).

If the weight of the whole system is 100, the separate weight values for each SPU are:

- Weight of SPU1 (W1) = 100*90*50/(50*90+80*60) = 48
- Weight of SPU2 (W2) = 100*80*60/(50*90+80*60) = 52

For the incoming sessions, 48 percent of session are allocated to SPU1 while 52 percent of packets are allocated to SPU2.

The weighted numbers might take effect on the system within a short period before the central point checks the runtime usage information and adjusts the weights to a new value.

Release History Table

Release	Description
17.4R1	Starting with Junos OS Release 17.4R1, traffic is hashed and distributed to different SPUs by the IOC, based on a hash-based session distribution algorithm. This enhancement provides an even hash distribution among all SPUs by using a larger fixed-length hash table. In earlier Junos OS releases, the traffic distribution was uneven among all SPUs due to a fixed-length hash table.
15.1X49-D30	Starting in Junos OS Release 15.1X49-D30, the central point architecture is enhanced to handle higher concurrent sessions and connections per second (cps) for the SRX5000 Series device.
15.1X49-D30	Starting in Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, adaptive mode session distribution was replaced by enhancements to the central point architecture.
15.1X49-D10	Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, hash-based session distribution is the default mode for the SRX5400, SRX5600, and SRX5800 devices. Selection of hash keys depends on application protocols.

RELATED DOCUMENTATION

Flow-Based Sessions | 68
TCP Sessions | 75
ECMP Flow-Based Forwarding | 86

Fragmentation Packets with PowerMode IPsec

PowerMode IPsec (PMI) is a new mode of operation for SRX4100, SRX4200, SRX4600, SRX5400, SRX5600, SRX5800, and vSRX instances to improve IPsec performance. Starting with Junos OS Release 19.1R1, the PMI is enhanced to handle the incoming and outgoing fragment packets using first path or fast path processing.

Enable the PMI process by using the **set security flow power-mode-ipsec** command. To verify that the packets are leveraging PMI, use the show security flow pmi statistics command.

Understanding PMI First Path and Fast Path Processing

In a PMI first path processing:

- The incoming first path packet is delivered to flow to create session.
- The incoming fragment packets are delivered to flow for reassembling.
- The incoming packets are delivered to flow for advanced security service processing.

In a PMI fast path processing, the PMI driver is used:

- To encrypt and send out the incoming clear text directly.
- To decrypt and send out the incoming ESP packets directly with session match.

Switching between PMI First Path and Fast Path Processing

The first path processing involves more features and instructions, while the PMI fast path processing provides better performance. In a PMI session, the packet processing switches between first path and fast path based on the packets flow in the session.

- The PMI session with both fragment and non-fragment packets are processed by first path.

- When the session only has non-fragmented packets, the session will switch from first path to fast past processing.

> **NOTE:** On SRX5400, SRX5600, and SRX5800 devices, switching happens after the NP session timeout.

Fragmentation for Incoming IP Packets

To support fragmentation for incoming IP packets for PMI, following steps are used in first path:

- PMI transmits all the fragmented IP packets in a session to the flow module for processing.
- PMI transmits all the non-fragmented IP packets in the same session to the flow module for packet ordering.
- The Flow module completes reassembly of fragmented packets and transmits the packets back to PMI for encryption.

Fragmentation for Outgoing IP Packets

To support fragmentation for outgoing IP packets for PMI, following steps are used:

- PMI detects clear text packets that requires fragmentation during session lookup and delivers packets to the flow module.
- Flow module does fragmentation for outgoing packets.
- PMI encrypts the packets before transmitting them.

NP session support

On SRX4100, SRX4200, and vSRX devices, fragment and non-fragment packets are hashed to the same CPU core for processing. Hence, NP session is not supported.

On SRX4600 devices with SPC3, fragment and non-fragment packets are hashed to different CPU cores for processing. Hence, NP session is supported to deliver fragment or non-fragment packets to the same core for ordering.

> **NOTE:** If a PMI session or non-PMI session has no NP session installed due to limited NP session capacity then the packet ordering for this PMI session may not be available.

Release History Table

Release	Description
19.1R1	Starting with Junos OS Release 19.1R1, the PMI is enhanced to handle the incoming and outgoing fragment packets using first path or fast path processing.

Unified Policies Support for Flow

IN THIS SECTION

- Flow First Path for Unified Policies | **110**
- Understanding Flow Fast Path | **111**
- Configuring the Session Log for the Default Security Policy | **112**
- Configuring the Session Timeout for the Default Security Policy | **112**

Starting in Junos OS Release 18.2R1, unified policies are supported on SRX Series devices, allowing granular control and enforcement of dynamic Layer 7 applications within the security policy. Unified policies are the security policies that enable you to use dynamic applications as match conditions as part of the existing 5-tuple or 6-tuple (5-tuple with user firewall) match conditions to detect application changes over time.

Unified policies allow you to use dynamic application as a policy match criteria in the application. On applying Application Identification (AppID) to the traffic, the AppID checks several packets and identifies the application. After the application is identified, the final policy is applied to the session. The policy actions such as permit, deny, reject, or redirect are applied to the traffic as per the policy.

During the initial policy lookup phase, which occurs prior to a dynamic application being identified, if there are multiple policies n the potential policy list, the SRX Series device applies the default security policy until a more explicit match has occurred. The policy that best matches the application is the final policy.

For more information on unified policies, See [Unified Security Policies, Application Identification Support for Unified Policies, and Understanding IDP Policy Support for Unified Policies.]

Flow First Path for Unified Policies

When the device examines the first packet of a flow, it determines the corresponding security policy, and performs a security policy lookup. During this process following cases are observed:

- If the traffic matches a legacy security policy or the final policy, the session is created.
- If there are multiple policies in the potential policy list and there is a security policy conflict, then the default security policy is applied.
- If there are multiple policies in the potential policy list, and the policy action does not permit the traffic, then the session is closed. A log message is generated to indicate the reason for the session closure. The default security policy is required during policy conflict stage, because each policy in the potential policy list has different configuration values for MSS, TCP SYN check, session timeout interval, and so on. In this case, when the default security policy is applied, all the values configured in that policy are applied. When a default security policy is matched, the policy actions are applied for the session.

  NOTE:
  - The default security policy is system-defined policy. This policy cannot be deleted.
  - The default policy is created on every logical system level, similar to the global default policy.
  - The session timeout interval and session log values are leveraged from the default security policy and default values such as TCP-MSS and TCP SYN are leveraged from the flow configuration.

- When a default policy is applied, a potential metadata for the policy action is allocated. The potential metadata is updated according to the potential policy list.

NOTE:

- Having a default security policy helps in resolving in the potential policy list.
- There can be many sessions matching the default security policy; however, the application services defined in the policy for the permitted traffic can be different. The security flow information for each session is saved.
- When an SRX Series device is operating in chassis cluster mode, the information is synchronized from the primary node to the secondary node along with the flow session and the chassis cluster real time objects (RTO).

- When the final application is identified, the security policy matching with the final application is applied. The subsequent packets are processed according to the final policy.

Understanding Flow Fast Path

After the first packet in a flow has traversed the device and a session has been established for it, it undergoes fast path processing. When the device examines a security flow session with default policy, it performs a security policy lookup and following cases are observed:

- If the existing Application Identification requires an update, the policy lookup process in repeated. The process is repeated until an explicit policy is returned and replaced in the security flow session. If an implicit policy is returned, the traffic is denied and the session is closed.
- When the final application is identified, the final policy matching the traffic is applied. If the policy actions in the default and the final policy are similar, the final policy replaces the default policy in the security flow session. If the policy actions in the default and the final policy are different, default policy is retained and the security flow session is closed.

NOTE: When the final and the default policy with a deny action is matched, the security flow session is closed.

- To update a session, the session timeout, log, or counter configuration in the final policy is used.

Configuring the Session Log for the Default Security Policy

The default security policy is required to manage policy conflicts in the potential policy list. You can set the session logs for the required sessions in default security policy configurations:

You can enable logging at the end of a session and at the beginning of the session with the following commands:

1. Generate a Session_Create log for policies entering the global pre-id-default-policy.

```
[edit]
user@host# set security policies pre-id-default-policy then log session-init
```

> **CAUTION:** Configuring **session-init** logging for the **pre-id-default-policy** can generate a large amount of logs. Each session that enters the SRX that initially matches the **pre-id-default-policy** will generate an event. We recommend only using this option for troubleshooting purposes.

2. Generate a Session_Close log for policies which close without exiting the global pre-id-default-policy.

```
[edit]
user@host# set security policies pre-id-default-policy then log session-close
```

Configuring the Session Timeout for the Default Security Policy

You can set the session timeout for the required sessions in default security policy configurations. You can specify the timeout values for UDP, TCP, ICMP, and ICMP6 sessions using the **set security policies pre-id-default-policy then session-timeout** command:

- Specify the timeout value in seconds for the TCP session:

```
[edit]
user@host# set security policies pre-id-default-policy then session-timeout tcp 1200
```

- Specify the timeout value in seconds for the UDP session:

```
[edit]
user@host# set security policies pre-id-default-policy then session-timeout udp 60
```

- Specify the timeout value in seconds for the ICMP session:

```
[edit]
user@host# set security policies pre-id-default-policy then session-timeout icmp 60
```

- Specify the timeout value in seconds for the ICMP6 session:

```
[edit]
user@host# set security policies pre-id-default-policy then session-timeout icmp6 120
```

Release History Table

Release	Description
18.2R1	Starting in Junos OS Release 18.2R1, unified policies are supported on SRX Series devices, allowing granular control and enforcement of dynamic Layer 7 applications within the security policy. Unified policies are the security policies that enable you to use dynamic applications as match conditions as part of the existing 5-tuple or 6-tuple (5-tuple with user firewall) match conditions to detect application changes over time.

RELATED DOCUMENTATION

Flow-Based Sessions | 68

TAP Mode for Flow Sessions

IN THIS SECTION

- Understanding TAP Mode Support for Security Flow Sessions | 114
- Example: Configuring Security Flow Sessions in TAP mode | 114

In TAP mode, an SRX Series device will be connected to a mirror port of the switch, which provides a copy of the traffic traversing the switch. An SRX Series device in TAP mode processes the incoming traffic from TAP interface and generates a security log to display the information on threats detected, application usage, and user details.

Understanding TAP Mode Support for Security Flow Sessions

Starting in Junos OS Release 18.3R1, TAP mode supports security flow sessions. The security flow session configuration remains the same as non-TAP mode. When you configure an SRX Series device to operate in TAP mode, the device generates a security log information to display the information on threats detected, application usage, and user details according to the incoming traffic. In TAP mode, the SRX Series device receives packets only from the configured TAP interface.

Traffic with and without VLAN can be received by TAP interface. By default, on all SRX Series devices, the **FLOW SYN-check** and **sequence-check** options are disabled at **[set security]** hierarchy level.

Starting in Junos OS Release 20.1R1, TAP mode inspects IP-IP and GRE inner tunnel traffic by de-encapsulating the outer and inner IP header (up to two levels) to create flow sessions. You can configure up to eight TAP interfaces on an SRX Series device.

Example: Configuring Security Flow Sessions in TAP mode

IN THIS SECTION

- Requirements | **115**
- Overview | **115**
- Configuration | **115**
- Verification | **116**

This example shows how to configure security flow sessions when the SRX device is configured in TAP mode.

Requirements

This example uses the following hardware and software components:

- An SRX Series device
- Junos OS Release 19.1R1

Before you begin:

- Read the *SRX TAP Mode Support Overview* to understand how and where this procedure fits in the overall support for security flow sessions.

Overview

In this example, you configure the security flow sessions when the SRX device is configured in TAP mode. Sessions are created when a TCP SYN packet is received and permitted by the security policy.

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter commit from configuration mode.

```
set security flow tcp-session no-syn-check
set security flow tcp-session no-sequence-check
```

Step-by-Step Procedure

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

To configure security flow sessions in TAP mode:

1. Configure the security flow session.

```
user@host# set security flow tcp-session no-syn-check
user@host# set security flow tcp-session no-sequence-check
```

Results

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

```
[edit]
user@host# show security flow
   tcp-session {
      no-syn-check;
      no-sequence-check;
   }
```

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

Verification

IN THIS SECTION

- Verifying Security Session Configuration in TAP Mode | **116**

To confirm that the configuration is working properly, perform these tasks:

Verifying Security Session Configuration in TAP Mode

Purpose

Verify information about security sessions.

Action

From operational mode, enter the **show security flow session** command.

```
user@host> show security flow session
```

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC4 PIC0:
Total sessions: 0

Flow Sessions on FPC4 PIC1:
Total sessions: 0
```

Meaning

Displays information about all currently active security sessions on the device in TAP mode.

SEE ALSO

SRX TAP Mode Support Overview

Flow Management in SRX Series Devices Using VRF Routing Instance

IN THIS SECTION

- Virtual Routing and Forwarding Instances in SD-WAN Deployments | **117**
- Flow Management Using VRF Routing Instance | **118**
- Virtual Routing and Forwarding Groups | **119**
- Flow Processing using Virtual Routing and Forwarding Group | **124**

Virtual Routing and Forwarding Instances in SD-WAN Deployments

Virtual routing and forwarding (VRF) instances are required to separate the routes of each tenant from the route of other tenants and from other network traffic. SRX Series devices use VRF instances for segmenting networks for increased security and improved manageability in SD-WAN deployments. For example, you can create distinct routing domains called tenants to segment large corporate networks and segment traffic to support multiple customer networks. Each tenant has its own routing table, which enables the support for overlapping IP subnets. VRF can be used to manage routes and to forward traffic based on independent forwarding tables in VRF for a specific tenant.

In an SD-WAN deployments, a provider edge (PE) router can be both a hub device and a spoke device that receives and forwards MPLS traffic. A customer edge (CE) router is an SRX Series device that interacts with a PE router to transmit VPN traffic using VRF routing instances. The VRF instances forward each customer VPN traffic and each VRF instance contains one label to represent all the customer traffic that flows through that VRF.

Different sites that connect to a spoke-side SRX Series device can belong to the same tenant or to the same VRF routing instance. These sites send the IP traffic that is intended to reach either public Internet or remote tenant sites.

When the traffic reaches the spoke-side SRX Series device, the device identifies the VRF instance from the LAN interfaces that are connected to those sites. After security processing on this traffic, the traffic

finds a route to the destination in that VRF routing table. If the destination is MPLS over next-hop-based generic routing encapsulation (GRE), the SRX Series device adds a corresponding MPLS label and forwards the packet to the hub-side device.

At the hub-side device, after receiving MPLS over GRE tunneled traffic, the SRX Series device associates the MPLS label to identify the corresponding VRF routing instance. After security processing of the traffic is complete, the device identifies whether the destination is on public Internet or reachable via MPLS next-hop.

If the destination is public Internet, Network Address Translation (NAT) converts VRF private IP address to a public IP address and establish the session. If the destination is a type of MPLS next-hop, corresponding MPLS label is added and the packet is forwarded to the remote spoke using a GRE overlay tunnel.

At the remote spoke side, after receiving the MPLS over GRE tunnel traffic, the device identifies the corresponding VRF routing-instance using the MPLS labels. Using that VRF routing instance, the SRX Series device finds the correct destination LAN interface in that VRF to forward the packet to the destination.

Flow Management Using VRF Routing Instance

An SRX Series device flow creates sessions based on 5-tuple data (source IP address, destination IP address, source port number, destination port number, and protocol number) along with interface tokens of input interface and output interface of traffic. For example, the routing instance VRF-1 and the routing instance VRF-2 have the same 5-tuple traffic that can enter and exit through the same physical GRE tunnel. When these overlapping IP addresses from the same tunnel enter or exit through the SRX Series device, then SRX device flow cannot install multiple sessions in the database because of the conflict in session information. Additional information is required for the SRX Series device to differentiate sessions during the installation.

Starting in Junos OS Release 15.1X49-D160, SRX Series devices can use VRF information from the MPLS-tagged packets in the session key to differentiate sessions. To differentiate sessions from different VRF instances, flow uses VRF identification numbers to the existing session key to identify each VRF instance. This session key is used as one of the matching criteria during session look-up.

You can use the following matching criteria along with existing 5-tuple matching conditions in a security policy to permit or deny traffic based on given VRF:

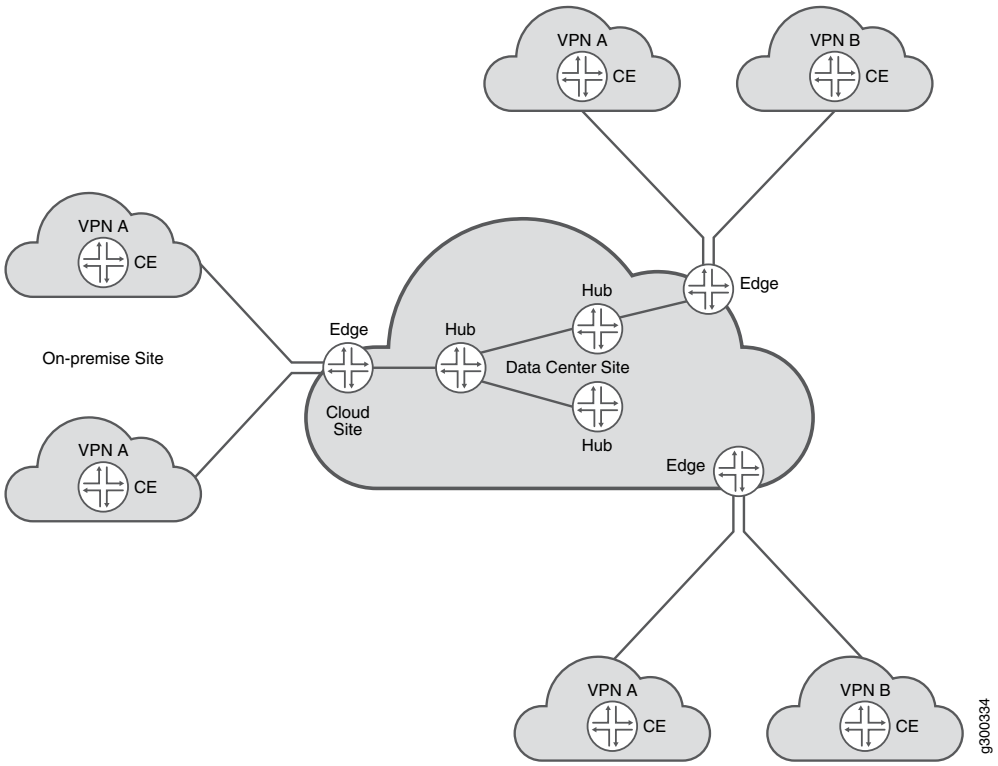
- Source VRF—This is the VRF routing instance associated with the incoming interface of the packet. When an incoming MPLS packet containing a label arrives at an SRX Series device, the device decodes the label, and maps the label to the incoming interface.
- Destination VRF—This is the VRF routing instance associated with the final route to the destination. During the first packet processing for a new session, flow needs a destination route for routing a packet to the next-hop device or interface. Flow searches the initial routing table from either the incoming

interface or from a separate RTT table until it finds the final next-hop device or interface. Once the final route entry is found, and if that route points to an MPLS next-hop device, then the destination VRF is assigned to the routing instance in which the final route is found.

Virtual Routing and Forwarding Groups

SD-WAN enterprise network is composed of multiple L3VPN networks as shown in Figure 11 on page 119.

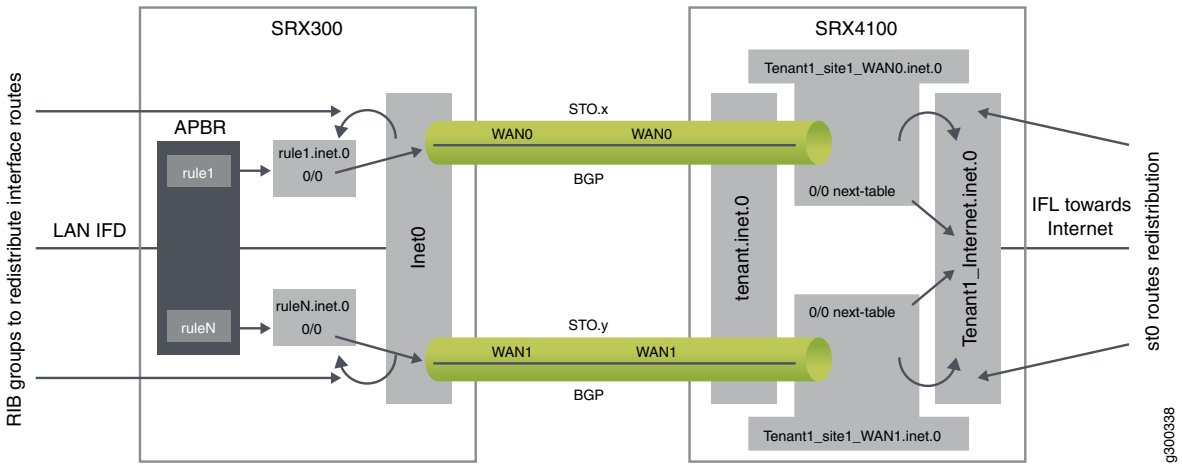
Figure 11: Multiple L3VPNs



L3VPN networks are identified at a site (CPE device) as a set of VRF instances. The VRF instances at a site belonging to a L3VPN network at a site are used for application policy based forwarding. SRX flow session handling has enhanced to support mid-stream traffic switching between these VRF instances based on application based steering policies. The VRF instances which are logically part of a given L3VPN network can be configured as a VRF group. Existing firewall, NAT configuration commands have been enhanced to support operations on VRF group.

The Figure 12 on page 120 describes how traffic steering within an L3VPN is done across multiple VRFs based on APBR policies.

Figure 12:



When you configure the VRF groups using VRF instances, a VRF group-ID is generated. These VRF groups are used in the following modules to control SD-WAN L3VPN:

- **Security Policy** - For policy control.
- **Flow** - To search policies based on VRF group names, along with source or destination zone, source or destination IP address, and protocol. Hence, sessions are created using VRF groups as one of differentiator.
- **NAT** - To support NAT rules based on VRF group names.
- **ALG** - To create ALG sessions using VRF groups as one of differentiator.

The functionality of the VRF groups:

- It allows a session to switch between two MPLS VRFs.
- When the VRF instances are part of the same VRF group, security features such as flow, policy, NAT, or ALG modules treat the VRF instances similarly.
- When you configure the VRF groups using VRF instances, a VRF group-ID is generated. This group-ID is stored in session for identifying the VRF group of a particular VRF instance.

Understanding VRF groups

VRF group is introduced to support L3VPN MPLS based sessions in SD-WAN network. It is used to control the MPLS L3VPN traffic in policy, flow, NAT and ALG modules when there are overlapping or no overlapping IP network addresses in the MPLS L3VPN network.

If the traffic pass between non MPLS L3VPN networks, VRF groups are not configured. When VRF groups are not configured, the VRF group-ID will be zero or the policy will use the option **any** for VRF group.

The purpose of VRF groups is:

- To differentiate L3VPN sessions between MPLS L3VPN network.
- To have policy and NAT control between MPLS L3VPN network.

Types of VRF groups

There are two important VRF group in L3VPN network are:

- Source-VRF group
- Destination-VRF group

To understand which VRF instances can be grouped together for Source-VRF group or Destination-VRF group, use the following information:

- **Source-VRF instances**—List of VRF instances that negotiates different MPLS paths to the same in-bound destination.
- **Destination-VRF instances**— List of VRF instances that contain the destination routes for a given L3VPN traffic.

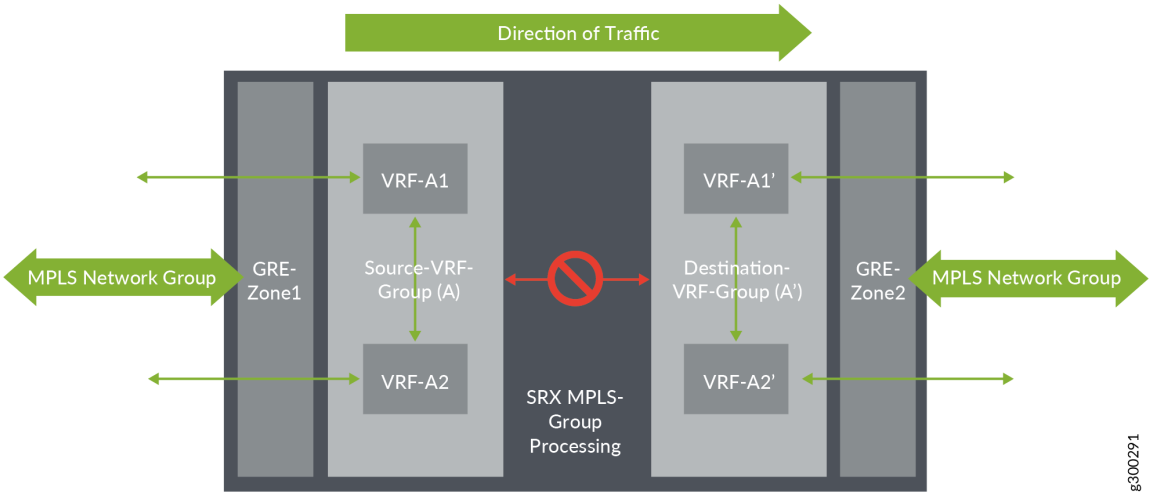
> **NOTE:** If the traffic is initiated in the opposite direction, the VRF groups switch roles with respect to the direction of the traffic.

VRF Movement

From Figure 13 on page 122, the initial traffic flow for a session establishment is from left to right. The traffic enters GRE-Zone1, then enters Source-VRF group (A) and passes through Destination-VRF group (A') before it exits through GRE_Zone2.

Similarly, the policy search is initiated from **GRE_Zone1->Source-VRF group(A)->Destination-VRF group->(A')->GRE_Zone2** and the flow sessions is set-up, using Source-VRF group (A) and Destination-VRF group (A) as an additional key values in sessions. When the flow sessions are done using VRF groups, traffic can switch (re-route) from one VRF to another VRF within Group.

Figure 13: VRF Movement within VRF Group



VRF group-ID

For storing the VRF group-ID, a 16-bits number is used in a session key data structure.

Configuring VRF groups

To configure a VRF group, use the following steps:

- List the VRF instances that needs to be grouped.
- Assign a name to the VRF group.
- Apply the VRF instances and the VRF group name in the CLI command **set security l3vpn vrf-group *group-name* vrf *vrf1* vrf *vrf2***

The source and destination VRF groups are configured separately based on different context.

- **Source VRF group**—The source VRF group for routing-instance is associated with MPLS packet. When the device receives a MPLS packet, the packet is decoded and mapped to LSI interface. The LSI interface contains the routing table information that helps in identifying the VRF group details.
- **Destination VRF group**—During first-path flow processing of packet for a new session, the destination route information is required to route the packet to the next-hop or interface. Flow searches the routing table to get the route information. When the received route information points to MPLS as next-hop, then the VRF of this route is used to identify the destination VRF group.

NOTE: The source and destination VRF groups are same in some cases when you prefer to control all the related VRFs in a L3VPN network.

VRF group Operations

When a VRF group is configured, a Group-ID is created which is unique for different VRF groups. You can perform different operations such as adding, removing, or modifying a VRF to a VRF group.

Adding VRF to a VRF group

When a VRF is added to a VRF group, the corresponding VRF group-ID is assigned to the VRF. When you add a VRF to VRF group, remember the following:

- A VRF can be added to only one VRF group. It cannot be a part of multiple VRF groups.
- A maximum of 32 VRFs are be configured in a VRF group.
- When a VRF is added, it impacts the existing session and a new session is created as per policy.
- When new sessions are created after adding a new VRF to VRF group, the sessions use the new VRF group-ID of the new VRF.

Removing VRF from a VRF group

When a VRF is removed from a VRF group, the VRF group-ID of that VRF group changes to zero but the VRF will still be available in the device. When you remove a VRF from a VRF group, it impacts the existing sessions in two ways:

- **Impacting existing sessions**—When a VRF is removed from the VRF group, the existing session is removed and a new session will be created as per policy
- **Match Traffic**—When a VRF is removed from VRF group, the VRF group-ID for that VRF changes to zero and hence will not match the session. The packet drops and a new session is created as per policy.

When a VRF is removed from the VRF group, the new session that is processed using the impacted VRF installs a new VRF group-ID. This VRF group-ID will be zero, or a new Group-ID is created if you add the VRF to a new VRF group

Modifying VRF group

Modifying a VRF group involves the following operations:

- **Changing VRF group name**: When you change the VRF group name, the policy module scans the existing sessions to verify if the new VRF group name matches the existing rules.
- **Adding VRF to VRF group**: When a VRF is added to a VRF group, the corresponding VRF group-ID is assigned to the VRF.

- **Removing VRF from VRF group**: When a VRF is removed from a VRF group, the VRF group-ID of that VRF changes to zero and still the VRF will be available in the device.

Removing VRF group

When you remove a VRF group using CLI, a session scan will be performed on the existing sessions to match the VRF group that is removed. If the session match the removed VRF group, then that session is removed from the device by setting an invalid timeout. For sessions that does not match the removed VRP-Group-ID are not impacted.

RELATED DOCUMENTATION

Flow Processing using Virtual Routing and Forwarding Group | 124

Flow Processing using Virtual Routing and Forwarding Group

IN THIS SECTION

- First Path Processing using VRF Group | 124
- Fast Path Processing using VRF Group | 126
- Example: Configuring a Security Policy to Permit VRF-Based Traffic from an IP Network to MPLS Network using VRF Group | 127
- Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to an IP Network using VRF Group | 131
- Example: Configuring a Security Policy to Permit VRF-Based Traffic from Public IP Network to MPLS Network using VRF Group | 135
- Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to Public IP Network to using VRF Group | 142
- Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to MPLS Network without NAT using VRF Group | 149
- Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to MPLS Network using NAT and VRF Group | 154

First Path Processing using VRF Group

To process a packet, the first path processing performs the following:

- **MPLS Decoder**—When flow receives a MPLS or non-MPLS packet, the packet is processed to retrieve the details of the incoming packet, interface, and routing-instance of the incoming interface.
- **FBF configuration**—When you configure FBF rules to re-direct the incoming packets to different routing-instance, the FBF rule finds the routing-instance information and pass the FBF routing-instance information instead of packet incoming interface routing-instance. This FBF VRF should be a part of VRF group to control the L3VPN network.
- **Initialize Routing-Table**—When the flow receives the packet, the initial routing-table for the packet is created. If the FBF configuration matches the firewall filters, then the routing-instance information from FBF is used for route look-up. Else, flow uses the incoming interface routing-instance information for route look-up.
- **Finding Source VRF group**—If the incoming packet is from MPLS network, then the packet is mapped to the VRF instance of source VRF group. If the incoming packet is not MPLS packet, then the source VRF group id is zero.
- **Destination NAT using VRF group**—Flow checks if the destination IP needs NAT translation. Destination NAT supports two types of match criteria for VRF:
  - NAT rule search using VRF routing-group.
  - NAT rule result using VRF routing-instance and NAT information.
- **Destination Route**—The route look-up which is done in initial route table is used to identify the outgoing interface and destination-VRF information. This information is used in policy search and session installation.
- **Final next-hop**—The first step in finding destination route is to find final the next-hop of the pointed route. Using this next-hop, flow will check if the next-hop points to MPLS network or not. If it is not pointing to MPLS network, the destination VRF group will be zero.
- **Destination VRF group**— When the destination VRF is identified, the destination VRF Group-ID is initialized. If the destination VRF is not assigned to any group, it is set to zero.
- **First Path Policy Search**—Flow performs policy search to check if the packet needs to be permitted or denied. Flow gathers the 5-tuple policy-key information and VRF information and this information is used by policy search module to find the appropriate VRF policy.
- **Source NAT using VRF group**—Flow session does source NAT using source VRF group NAT rule search. Source-NAT supports two types of NAT search criteria.
  - Source-NAT rule search using VRF group.
  - Static-NAT rule search using VRF group or VRF instance.
- **Static NAT using VRF group or VRF instance**—Static NAT supports routing-group in rule-set and routing-instance in rule with VRF type.
  - When static NAT matches as destination NAT translation for a given IP packet, the VRF routing-group will be one of the match criteria and the VRF routing-instance ill be used as destination routing table.

  - When static NAT matches as source NAT translation for a given IP packet, the VRF routing-instance will be one of the match criteria.

- **Session Installation using VRF group**—During session installation process, source VRF group-ID is stored in forward-wing indicating that the wing points MPLS network. The destination VRF group-ID that is found from route look-up is stored in reverse-wing indicating that the wing points MPLS network.
- **Re-routing using VRF group**—Once the session is established using VRF group information, re-route is initiated if the interface is down or initial route is not available. These changed routes should be part of same VRF group (Source-VRF group/Destination-VRF group), in which the session is initially established on either side. Else, traffic will not match session and future traffic of session might get dropped or create new sessions as per policy.

Fast Path Processing using VRF Group

The fast path processing performs the following steps to process a packet.

- **MPLS Decoder**—When a packet MPLS or non-MPLS packet is received, the packet undergoes MPLS processing. When the processing is complete, the flow receives the details of the incoming packet, interface, and routing-instance of the incoming interface.
- **FBF configuration**—When you configure FBF rules to re-direct the incoming packets to different routing-instance, the FBF rule finds the routing-instance information and pass the FBF routing-instance information instead of packet incoming interface routing-instance. This FBF VRF should be a part of VRF group to control the L3VPN network.
- **Session look-up using VRF Group-ID**—During session look-up process, flow checks whether to pass the VRF Group-ID in session key for look-up. If the incoming interface is MPLS, flow will pass the VRF Group-ID information of the mapped VRF routing-instance to session key along with other key tuple information. If the incoming interface is not MPLS, the VRF Group-ID will be zero.
- **Session Route change**—If the route changes for session in mid-stream, flow checks for the new VRF that belongs to this route. If the new VRF Group-ID differs from the VRF Group-ID of the session, then the route will not be processed and the future packets are dropped. Hence, for re-routing the new route should belong to a VRF that belongs to session VRF Group.
- **VRF Group policy change**—When VRF group session policy is changed due to policy attributes such as zone/interface/IP/Source-VRF group/Destination-VRF group, the policy will be re-matched for the same session by supplying policy 5-tuple along with source VRF group and destination VRF group values to check if the policy is valid or not. Upon re-match, if policy does not match the session information, then the session terminates.
- **VRF session display**—Source-VRF Group and Destination-VRF Group are displayed in session output display to differentiate different VRF group for the same tuple.
- **High Availability**—High availability is supported with no behavior change when additional VRF group-ID information is synchronized to the HA peer node for differentiating different VRF group in the session

Example: Configuring a Security Policy to Permit VRF-Based Traffic from an IP Network to MPLS Network using VRF Group

IN THIS SECTION

- Requirements | 127
- Overview | 127
- Configuration | 128

This example shows how to configure a security policy to permit traffic from a private IP network to MPLs network using VRF group.

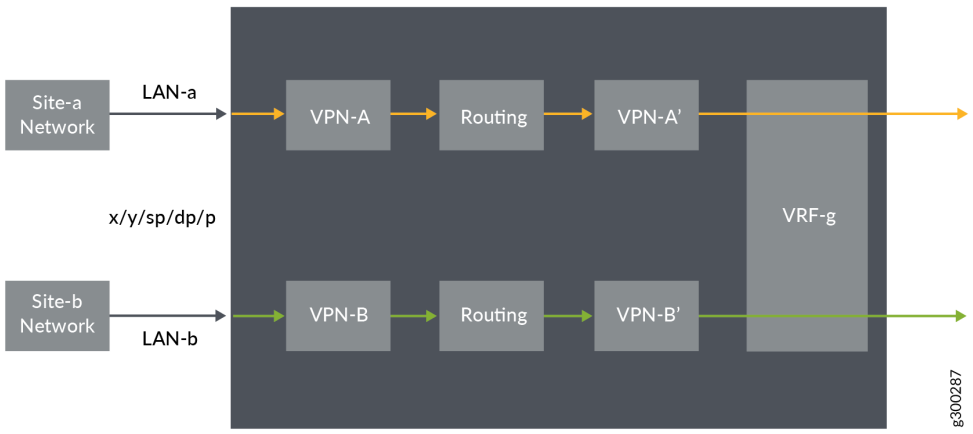
Requirements

- Understand how to create a security zone. See *Example: Creating Security Zones*.
- Supported SRX Series device with Junos OS Release 15.1X49-D170 or later. This configuration example is tested for Junos OS Release 15.1X49-D170.

Overview

In Junos OS, security policies enforce rules for transit traffic, in terms of what traffic can pass through the device and the actions that need to take place on the traffic as it passes through the device. In Figure 14 on page 127, an SRX Series device is deployed in an SD-WAN to permit the traffic from a private IP network to MPLS network using VRF group.

Figure 14: Traffic from Private Network to MPLS



This configuration example shows how to:

- Permit traffic from IP network (LAN-a) to VRF group
- Permit traffic from IP network (LAN-b) to VRF group

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security l3vpn vrf-group vpn-A vrf VRF-A1
set security l3vpn vrf-group vpn-A vrf VRF-A2
set security l3vpn vrf-group vpn-A1 vrf VRF-A11
set security l3vpn vrf-group vpn-A1 vrf VRF-A21
set security l3vpn vrf-group vpn-B vrf VRF-B1
set security l3vpn vrf-group vpn-B vrf VRF-B2
set security l3vpn vrf-group vpn-B1 vrf VRF-B11
set security l3vpn vrf-group vpn-B1 vrf VRF-B21
set security policies from-zone LAN-a_Zone to-zone GRE_Zone policy vrf-a_policy match source-address any
set security policies from-zone LAN-a_Zone to-zone GRE_Zone policy vrf-a_policy match destination-address
   any
set security policies from-zone LAN-a_Zone to-zone GRE_Zone policy vrf-a_policy match application any
set security policies from-zone LAN-a_Zone to-zone GRE_Zone policy vrf-a_policy match
   destination-l3vpn-vrf-group vpn-A1
set security policies from-zone LAN-a_Zone to-zone GRE_Zone policy vrf-a_policy then permit
set security policies from-zone LAN-b_Zone to-zone GRE_Zone policy vrf-b_policy match source-address any
set security policies from-zone LAN-b_Zone to-zone GRE_Zone policy vrf-b_policy match destination-address
   any
set security policies from-zone LAN-b_Zone to-zone GRE_Zone policy vrf-b_policy match application any
set security policies from-zone LAN-b_Zone to-zone GRE_Zone policy vrf-b_policy match
   destination-l3vpn-vrf-group vpn-B1
set security policies from-zone LAN-b_Zone to-zone GRE_Zone policy vrf-b_policy then permit
```

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*.

1. Create VRF group vpn-A with VRF instances A1 and A2

```
[edit security]
user@host# set l3vpn vrf-group vpn-A vrf VRF-A1
user@host# set l3vpn vrf-group vpn-A vrf VRF-A2
```

2. Create VRF group vpn-A1 with VRF instances A11, and A21

```
[edit security]
user@host# set l3vpn vrf-group vpn-A1 vrf VRF-A11
user@host# set l3vpn vrf-group vpn-A1 vrf VRF-A21
```

3. Create VRF group vpn-B with VRF instances B1 and B2

```
[edit security]
user@host# set l3vpn VRF group vpn-B vrf VRF-B1
user@host# set l3vpn VRF group vpn-B vrf VRF-B2
```

4. Create VRF group vpn-B1 with VRF instances B11 and B21

```
[edit security]
user@host# set l3vpn vrf-group vpn-B1 vrf VRF-B11
user@host# set l3vpn vrf-group vpn-B1 vrf VRF-B21
```

5. Create a security policy to permit vrf-a traffic.

```
[edit security policies from-zone LAN-a_Zone to-zone GRE_Zone]
user@host# set policy vrf-a_policy match source-address any
user@host# set policy vrf-a_policy match destination-address any
user@host# set policy vrf-a_policy match application any
user@host# set policy vrf-a_policy match destination-l3vpn-vrf-group vpn-A1
user@host# set policy vrf-a_policy then permit
```

6. Create a security policy to permit vrf-b traffic.

```
[edit security policies from-zone LAN-a_Zone to-zone GRE_Zone]
user@host# set policy vrf-b_policy match source-address any
user@host# set policy vrf-b_policy match destination-address any
user@host# set policy vrf-b_policy match application any
user@host# set policy vrf-b_policy match destination-l3vpn-vrf-group vpn-B1
user@host# set policy vrf-b_policy then permit
```

Results

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

```
[edit]
user@host# show security policies
from-zone LAN-a_Zone to-zone GRE_Zone {
    policy vrf-a_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            source-l3vpn-VRF group vpn-A1;
        }
        then {
            permit;
        }
    }
from-zone LAN-b_Zone to-zone GRE_Zone {
    policy vrf-b_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            source-l3vpn-VRF group vpn-B1;
        }
        then {
            permit;
        }
        }
    }
```

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

Verification

Verifying Policy Configuration

Purpose

Verify information about security policies.

Action

From operational mode, enter the **show security policies** command to display a summary of all the security policies configured on the device.

```
user@root> show security policies
Default policy: permit-all
From zone: LAN-a_Zone, To zone: GRE_Zone
  Policy: vrf-a_policy, State: enabled, Index: 4, Scope Policy: 0, Sequence number:
```

```
 1
    Source L3VPN vrf-group: vpn-A1
    destination L3VPN VRF Group: any
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
From zone: LAN-b_Zone, To zone: GRE_Zone
  Policy: vrf-b_policy, State: enabled, Index: 5, Scope Policy: 0, Sequence number:
 2
    Source L3VPN vrf-group: vpn-B1
    destination L3VPN VRF Group: any
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to an IP Network using VRF Group

IN THIS SECTION

- Requirements | **131**
- Overview | **132**
- Configuration | **132**

This example shows how to configure a security policy to permit traffic from MPLS to IP netwrok using the VRF group.

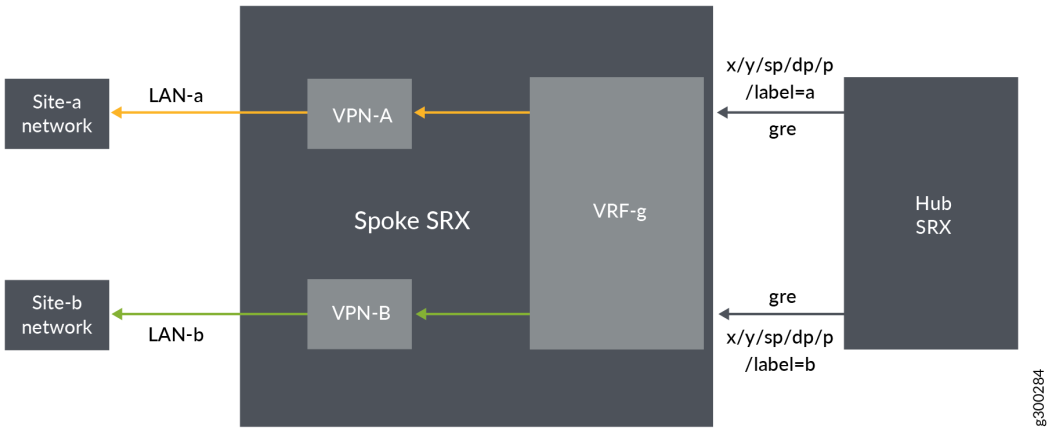
Requirements

- Understand how to create a security zone. See *Example: Creating Security Zones*.
- Supported SRX Series device with Junos OS Release 15.1X49-D170 or later. This configuration example is tested for Junos OS Release 15.1X49-D170.
- Configure network interfaces on the device. See *Interfaces User Guide for Security Devices*.

Overview

In Junos OS, security policies enforce rules for transit traffic, in terms of what traffic can pass through the device and the actions that need to take place on the traffic as it passes through the device. In Figure 15 on page 132, an SRX Series device is deployed in an SD-WAN to permit traffic from a MPLS network to private network using VRF group.

Figure 15: Traffic Permit from MPLS to Private Network



This configuration example shows how to:

- Permit traffic from GRE MPLS to LAN-a
- Permit traffic from GRE MPLS to LAN-b

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security l3vpn vrf-group vpn-A vrf VRF-A1
set security l3vpn vrf-group vpn-A vrf VRF-A2
set security l3vpn vrf-group vpn-B vrf VRF-B1
set security l3vpn vrf-group vpn-B vrf VRF-B2
set security policies from-zone GRE_Zone to-zone LAN-a_Zone policy vrf-a_policy match source-address any
set security policies from-zone GRE_Zone to-zone LAN-a_Zone policy vrf-a_policy match destination-address
    any
set security policies from-zone GRE_Zone to-zone LAN-a_Zone policy vrf-a_policy match application any
```

```
set security policies from-zone GRE_Zone to-zone LAN-a_Zone policy vrf-a_policy match source-l3vpn-vrf-group
    vpn-A
set security policies from-zone GRE_Zone to-zone LAN-a_Zone policy vrf-a_policy then permit
set security policies from-zone GRE_Zone to-zone LAN-b_Zone policy vrf-b_policy match source-address any
set security policies from-zone GRE_Zone to-zone LAN-b_Zone policy vrf-b_policy match destination-address
    any
set security policies from-zone GRE_Zone to-zone LAN-b_Zone policy vrf-b_policy match application any
set security policies from-zone GRE_Zone to-zone LAN-b_Zone policy vrf-b_policy match source-l3vpn-vrf-group
    vpn-B
set security policies from-zone GRE_Zone to-zone LAN-b_Zone policy vrf-b_policy then permit
```

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*.

1. Create VRF group vpn-A with VRF instances A1 and A2.

```
[edit security]
user@host# set l3vpn vrf-group vpn-A vrf VRF-A1
user@host# set l3vpn vrf-group vpn-A vrf VRF-A2
```

2. Create VRF group vpn-B with VRF instances B1 and B2.

```
[edit security]
user@host# set l3vpn vrf-group vpn-B vrf VRF-B1
user@host# set l3vpn vrf-group vpn-B vrf VRF-B2
```

3. Create a security policy to permit VRF-a traffic.

```
[edit security policies from-zone GRE_Zone to-zone LAN-a_Zone]
user@host# set policy vrf-a_policy match source-address any
user@host# set policy vrf-a_policy match destination-address any
user@host# set policy vrf-a_policy match application any
user@host# set policy vrf-a_policy match source-l3vpn-vrf-group vpn-A
user@host# set policy vrf-a_policy then permit
```

4. Create a security policy to permit VRF-b traffic.

```
[edit security policies from-zone GRE_Zone to-zone LAN-b_Zone]
user@host# set policy vrf-b_policy match source-address any
```

```
user@host# set policy vrf-b_policy match destination-address any
user@host# set policy vrf-b_policy match application any
user@host# set policy vrf-b_policy match destination-l3vpn-vrf-group vpn-B
user@host# set policy vrf-b_policy then permit
```

Results

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

```
[edit]
user@host# show security policies
from-zone GRE_Zone to-zone LAN-a_Zone {
    policy vrf-a_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            destination-l3vpn-vrf-group vpn-A;
        }
        then {
            permit;
        }
    }
from-zone GRE_Zone to-zone LAN-b_Zone {
    policy vrf-b_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            destination-l3vpn-vrf-group vpn-B;
        }
        then {
            permit;
        }
    }
}
```

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

Verification

Verifying Policy Configuration

Purpose

Verify information about security policies.

Action

From operational mode, enter the **show security policies** command to display a summary of all the security policies configured on the device.

```
user@root> show security policies
Default policy: permit-all
From zone: GRE_Zone, To zone: LAN-a_Zone
  Policy: vrf-a_policy, State: enabled, Index: 4, Scope Policy: 0, Sequence number:
 1
    Source L3VPN VRF-Group: any
    destination L3VPN VRF Group: vpn-A
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
From zone: GRE_Zone, To zone: LAN-b_Zone
  Policy: vrf-b_policy, State: enabled, Index: 5, Scope Policy: 0, Sequence number:
 2
    Source L3VPN VRF Group: any
    destination L3VPN VRF-Group: vpn-B
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

Example: Configuring a Security Policy to Permit VRF-Based Traffic from Public IP Network to MPLS Network using VRF Group

IN THIS SECTION

- Requirements | 136
- Overview | 136
- Configuration | 136
- Verification | 140

This example describes how to configure the destination NAT rule to translate incoming public IP network to MPLS network using VRF group.

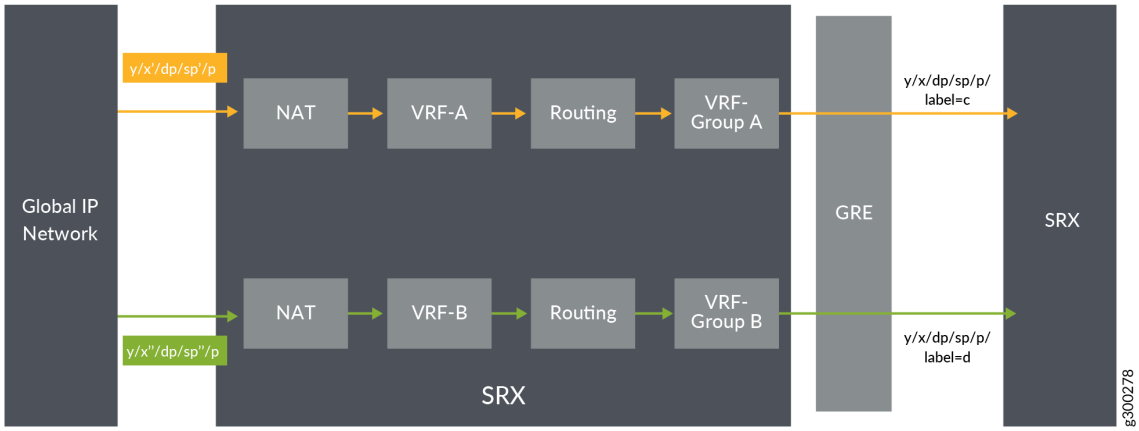
Requirements

- Understand how SRX Series devices work in an SD-WAN deployment for NAT.
- Understand Virtual Routing and Forwarding Instances. See "Virtual Routing and Forwarding Instances in SD-WAN Deployments" on page 117.

Overview

In Figure 16 on page 136, an SRX Series device is configured with destination NAT rule to translate incoming public IP network to per VRF based destination routing table and IP. The SRX Series device is configured with two VRF groups, vpn-A and vpn-B.

Figure 16: Traffic Permit from Public Network to MPLS



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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security l3vpn vrf-group vpn-A vrf VRF-A1
set security l3vpn vrf-group vpn-A vrf VRF-A2
set security l3vpn vrf-group vpn-B vrf VRF-B1
set security l3vpn vrf-group vpn-B vrf VRF-B2
set security nat destination pool vrf-a_p routing-instance VRF-a
set security nat destination pool vrf-a_p address 192.168.1.200
```

```
set security nat destination rule-set rs from interface ge-0/0/1.0
set security nat destination rule-set rs rule vrf-a_r match destination-address 203.0.113.200
set security nat destination rule-set rs rule vrf-a_r then destination-nat pool vrf-a_p
set security nat destination pool vrf-b_p routing-instance VRF-b
set security nat destination pool vrf-b_p address 192.168.1.201
set security nat destination rule-set rs from interface ge-0/0/1.1
set security nat destination rule-set rs rule vrf-b_r match destination-address 203.0.113.201
set security nat destination rule-set rs rule vrf-b_r then destination-nat pool vrf-b_p
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-a_policy match source-address any
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-a_policy match destination-address any
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-a_policy match application any
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-a_policy match source-l3vpn-vrf-group
    vpn-A
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-a_policy then permit
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-b_policy match source-address any
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-b_policy match destination-address any
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-b_policy match application any
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-b_policy match source-l3vpn-vrf-group
    vpn-B
set security policies from-zone GE_Zone to-zone GRE_Zone policy vrf-b_policy then permit
```

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 destination NAT mapping for a single VRF:

1. In Layer 3 VPNs create a VRF group vpn-A with VRF instances A1 and A2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-A vrf VRF-A1
user@host#set l3vpn vrf-group vpn-A vrf VRF-A2
```

2. Create another VRF group vpn-B with VRF instances B1 and B2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-B vrf VRF-B1
user@host#set l3vpn vrf-group vpn-B vrf VRF-B2
```

3. Specify a destination NAT IP address pool.

```
[edit security nat destination]
```

```
user@host# set pool vrf-a_p address 192.168.1.200
user@host# set pool vrf-b_p address 192.168.1.201
```

4. Assign the routing instance to the destination pool.

```
[edit security nat destination]
user@host# set pool vrf-a_p routing-instance VRF-a
user@host# set pool vrf-b_p routing-instance VRF-b
```

5. Create a destination NAT rule set.

```
[edit security nat destination]
user@host# set rule-set rs from routing-group vpn-A
user@host# set rule-set rs from routing-group vpn-B
user@host# set rule-set rs from interface ge-0/0/1.0
user@host# set rule-set rs from interface ge-0/0/1.1
```

6. Configure a rule that matches packets and translates the destination IP address to an IP address in the destination NAT IP address pool.

```
[edit security nat destination]
user@host# set rule-set rs rule vrf-a_r match destination-address 203.0.113.200
user@host# set rule-set rs rule vrf-a_r then destination-nat pool vrf-a_p
user@host# set rule-set rs rule vrf-b_r match destination-address 203.0.113.201
user@host# set rule-set rs rule vrf-b_r then destination-nat pool vrf-b_p
```

7. Create a security policy to permit VRF-a traffic.

```
[edit security policies from-zone GE_Zone to-zone GRE_Zone]
user@host# set policy vrf-a_policy match source-address any
user@host# set policy vrf-a_policy match destination-address any
user@host# set policy vrf-a_policy match application any
user@host# set policy vrf-a_policy match destination-l3vpn-vrf-group vpn-A
user@host# set policy vrf-a_policy then permit
```

8. Create a security policy to permit VRF-b traffic.

```
[edit security policies from-zone GE_Zone to-zone GRE_Zone]
user@host# set policy vrf-b_policy match source-address any
```

```
user@host# set policy vrf-b_policy match destination-address any
user@host# set policy vrf-b_policy match application any
user@host# set policy vrf-b_policy match destination-l3vpn-vrf-group vpn-B
user@host# set policy vrf-b_policy then permit
```

Results

From configuration mode, confirm your configuration by entering the **show security nat** and **show security policies** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security nat
   destination {
      pool vrf-a_p {
         routing-instance {
            VRF-a;
         }
         address 192.168.1.200/32;
      }
      pool vrf-b_p {
         routing-instance {
            VRF-b;
         }
         address 192.168.1.201/32;
      }
      rule-set rs {
         from interface [ ge-0/0/1.0 ge-0/0/1.1 ];
         rule vrf-a_r {
            match {
               destination-address 203.0.113.200/32;
            }
            then {
               destination-nat {
                  pool {
                     vrf-a_p;
                  }
               }
            }
         }
         rule vrf-b_r {
            match {
               destination-address 203.0.113.201/32;
```

```
            }
            then {
               destination-nat {
                  pool {
                     vrf-b_p;
                  }
               }
            }
         }
      }
   }
```

```
[edit]
user@host# show security policies
from-zone GE_Zone to-zone GRE_Zone {
   policy vrf-a_policy {
      match {
         source-address any;
         destination-address any;
         application any;
         destination-l3vpn-vrf-group vpn-A;
      }
      then {
         permit;
      }
   }
   policy vrf-b_policy {
      match {
         source-address any;
         destination-address any;
         application any;
         destination-l3vpn-vrf-group vpn-B;
      }
      then {
         permit;
      }
   }
```

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

Verification

Verifying Destination NAT Rule Usage and Security Policies

Purpose

Verify that there is traffic matching the destination NAT rule.

Action

From operational mode, enter the **show security nat destination rule all** command. In the Translation hits field, verify whether there is traffic that matches the destination NAT rule.

```
user@host> show security nat destination rule all
Total destination-nat rules: 2
Total referenced IPv4/IPv6 ip-prefixes: 2/0
Destination NAT rule: vrf-a_r               Rule-set: rs
  Rule-Id                   : 1
  Rule position             : 1
  From interface           : ge-0/0/1.0
  Destination addresses     : 203.0.113.200   - 203.0.113.200
  Action                    : vrf-a_p
  Translation hits          : 0
    Successful sessions     : 0
    Failed sessions         : 0
  Number of sessions        : 0
    Destination NAT rule   : vrf-b_r
  Rule-set          : rs
  Rule-Id                   : 2
  Rule position             : 2
  From interface            : ge-0/0/1.1
  Destination addresses     : 203.0.113.201   - 203.0.113.201
  Action                    : vrf-b_p
  Translation hits          : 0
    Successful sessions     : 0
    Failed sessions         : 0
  Number of sessions        : 0
```

From operational mode, enter the **show security policies** command to display a summary of all the security policies configured on the device.

```
user@root> show security policies
Default policy: permit-all
From zone: GE_Zone, To zone: GRE_Zone
  Policy: vrf-a_policy, State: enabled, Index: 4, Scope Policy: 0, Sequence number:
 1
    Source L3VPN VRF Group: any
    destination L3VPN VRF-Group: vpn-A
    Source addresses: any
    Destination addresses: any
```

```
    Applications: any
    Action: permit
From zone: GE_Zone, To zone: GRE_Zone
  Policy: vrf-b_policy, State: enabled, Index: 5, Scope Policy: 0, Sequence number:
 2
    Source L3VPN VRF Group: any
    destination L3VPN VRF-Group: vpn-B
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to Public IP Network to using VRF Group

IN THIS SECTION

- Requirements | **142**
- Overview | **142**
- Configuration | **143**
- Verification | **147**

This example describes how to configure the routing group to translate per VRF group network traffic to global IP pool.

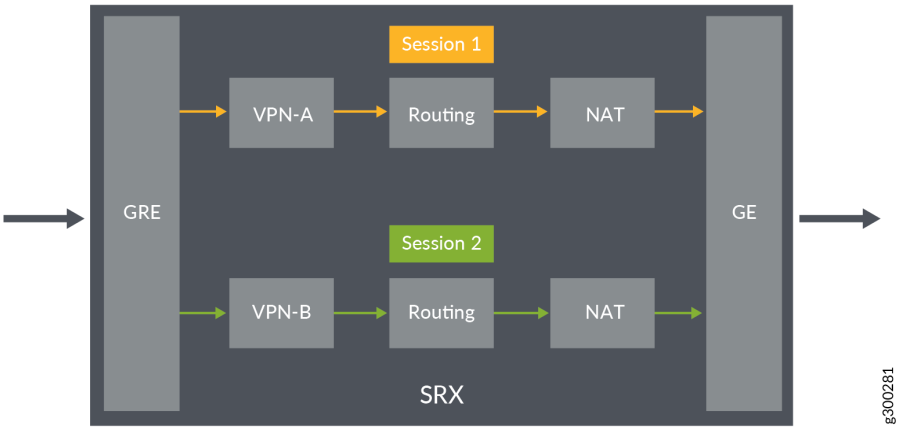
Requirements

- Understand how SRX Series devices work in an SD-WAN deployment for NAT.
- Understand Virtual Routing and Forwarding Instances. See “Virtual Routing and Forwarding Instances in SD-WAN Deployments” on page 117.

Overview

In Figure 17 on page 143, an SRX Series device is configured with routing group to permit VRF group network traffic from MPLS to global IP pool. The SRX Series device is configured with two VRF groups, vpn-A and vpn-B.

Figure 17: Traffic Permit from MPLS to Public Network



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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security l3vpn vrf-group vpn-A vrf VRF-A1
set security l3vpn vrf-group vpn-A vrf VRF-A2
set security l3vpn vrf-group vpn-B vrf VRF-B1
set security l3vpn vrf-group vpn-B vrf VRF-B2
set security nat source pool vrf-a_p address 203.0.113.200
set security nat source rule-set vrf-a_rs from routing-group vpn-A
set security nat source rule-set vrf-a_rs to zone GE_Zone
set security nat source rule-set vrf-a_rs rule rule1 match source-address 192.168.1.200
set security nat source rule-set vrf-a_rs rule rule1 then source-nat pool vrf-a_p
set security nat source pool vrf-b_p address 203.0.113.201
set security nat source rule-set vrf-b_rs from routing-group vpn-B
set security nat source rule-set vrf-b_rs to zone GE_Zone
set security nat source rule-set vrf-b_rs rule rule2 match source-address 192.168.1.201
set security nat source rule-set vpn-b_rs rule rule2 then source-nat pool vrf-b_p
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-a_policy match source-address any
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-a_policy match destination-address any
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-a_policy match application any
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-a_policy match source-l3vpn-vrf-group
    vpn-A
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-a_policy then permit
```

```
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-b_policy match source-address any
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-b_policy match destination-address any
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-b_policy match application any
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-b_policy match source-l3vpn-vrf-group
    vpn-B
set security policies from-zone GRE_Zone to-zone GE_Zone policy vrf-b_policy then permit
```

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 source NAT mapping for a single VRF:

1. In Layer 3 VPNs create a VRF group vpn-A with VRF instances A1 and A2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-A vrf VRF-A1
user@host#set l3vpn vrf-group vpn-A vrf VRF-A2
```

2. Create another VRF group vpn-B with VRF instances B1 and B2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-B vrf VRF-B1
user@host#set l3vpn vrf-group vpn-B vrf VRF-B2
```

3. Specify a source NAT IP address pool.

```
[edit security nat source]
user@host# set pool vrf-a_p address 192.168.1.200
user@host# set pool vrf-b_p address 192.168.1.201
```

4. Create a source NAT rule set.

```
[edit security nat source]
user@host# set rule-set rs from routing-group vpn-A
user@host# set rule-set rs from routing-group vpn-B
user@host# set rule-set rs to zone GE_Zone
```

5. Configure a rule that matches packets and translates per VRF group network traffic to global IP pool.

```
[edit security nat source]
user@host# set rule-set rs rule vrf-a_r match destination-address 203.0.113.200
user@host# set rule-set rs rule vrf-a_r then destination-nat pool vrf-a_p
user@host# set rule-set rs rule vrf-b_r match destination-address 203.0.113.201
user@host# set rule-set rs rule vrf-b_r then destination-nat pool vrf-b_p
```

6. Create a security policy to permit vpn-A traffic.

```
[edit security policies from-zone GRE_Zone to-zone GE_Zone]
user@host# set policy vrf-a_policy match source-address any
user@host# set policy vrf-a_policy match destination-address any
user@host# set policy vrf-a_policy match application any
user@host# set policy vrf-a_policy match destination-l3vpn-vrf-group vpn-A
user@host# set policy vrf-a_policy then permit
```

7. Create a security policy to permit vpn-B traffic.

```
[edit security policies from-zone GRE_Zone to-zone GE_Zone]
user@host# set policy vrf-b_policy match source-address any
user@host# set policy vrf-b_policy match destination-address any
user@host# set policy vrf-b_policy match application any
user@host# set policy vrf-b_policy match destination-l3vpn-vrf-group vpn-B
user@host# set policy vrf-b_policy then permit
```

Results

From configuration mode, confirm your configuration by entering the **show security nat** and **show security policies** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security nat
    source {
        pool vrf-a_p {
            address {
                203.0.113.200/32;
            }
        }
        pool vrf-b_p {
            address {
                203.0.113.201/32;
```

```
        }
      }
      rule-set vrf-a_rs {
        from routing-group vpn-A;
        to zone GE_Zone1;
        rule rule1 {
          match {
            source-address 192.168.1.200/32;
          }
          then {
            source-nat {
              pool {
                vrf-a_p;
              }
            }
          }
        }
        rule-set vrf-b_rs {
          from routing-group vpn-B;
          to zone GE_Zone;
          match {
            source-address 192.168.1.201/32;
          }
          then {
            source-nat {
              pool {
                vrf-b_p;
              }
            }
          }
        }
      }
    }
```

```
[edit]
user@host# show security policies
from-zone GRE_Zone to-zone GE_Zone {
    policy vrf-a_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            destination-l3vpn-vrf-group vpn-A;
        }
```

```
        then {
            permit;
        }
    }
    policy vrf-b_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            destination-l3vpn-vrf-group vpn-B;
        }
        then {
            permit;
        }
    }
}
```

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

Verification

Verifying Destination NAT Rule Usage and Security Policies

Purpose

Verify that there is traffic matching the source NAT rule.

Action

From operational mode, enter the **show security nat source rule all** command. In the Translation hits field, verify whether there is traffic that matches the source NAT rule.

```
user@host> show security nat source rule all
Total source-nat rules: 2
Total referenced IPv4/IPv6 ip-prefixes: 2/0
Source NAT rule      : vrf-a_r
Rule-set: rs
  Rule-Id                   : 1
  Rule position             : 1
  From routing-group        : vpn-A
  To zone                   : GE_Zone1
  Source addresses         : 203.0.113.200   - 203.0.113.200
  Action                    : vrf-a_p
  Translation hits          : 0
    Successful sessions     : 0
    Failed sessions         : 0
  Number of sessions        : 0
```

```
Source NAT rule       : vrf-b_r
Rule-set: rs
  Rule-Id                   : 2
  Rule position             : 2
  From routing-group        : vpn-A
  To zone                   : GE_Zone
  Destination addresses     : 203.0.113.201   - 203.0.113.201
  Action                    : vrf-b_p
  Translation hits          : 0
    Successful sessions     : 0
    Failed sessions         : 0
  Number of sessions        : 0
```

From operational mode, enter the **show security policies** command to display a summary of all the security policies configured on the device.

```
user@root> show security policies
Default policy: permit-all
From zone: GRE_Zone, To zone: GE_Zone
  Policy: vrf-a_policy, State: enabled, Index: 4, Scope Policy: 0, Sequence number:
 1
    Source L3VPN VRF Group: any
    destination L3VPN VRF Group: vpn-A
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
From zone: GRE_Zone, To zone: GE_Zone
  Policy: vrf-b_policy, State: enabled, Index: 5, Scope Policy: 0, Sequence number:
 2
    Source L3VPN VRF Group: any
    destination L3VPN VRF Group: vpn-B
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to MPLS Network without NAT using VRF Group

IN THIS SECTION

- Requirements | 149
- Overview | 149
- Configuration | 150
- Verification | 153

This example describes how to configure the routing group to permit traffic between MPLS networks without using NAT.

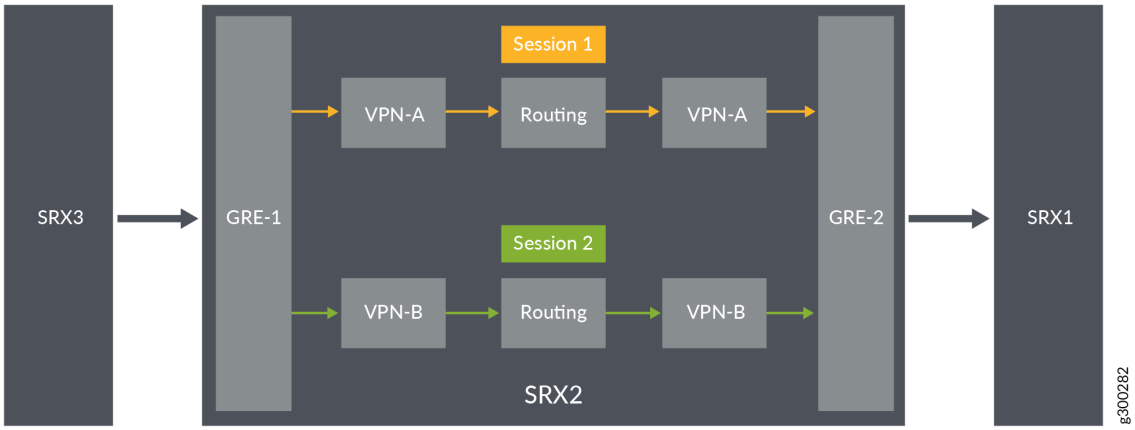
Requirements

- Understand how SRX Series devices work in an SD-WAN deployment for NAT.
- Understand Virtual Routing and Forwarding Instances. See "Virtual Routing and Forwarding Instances in SD-WAN Deployments" on page 117.

Overview

In Figure 18 on page 150, an SRX Series device is configured with routing group to permit traffic between MPLS networks without using NAT. The SRX Series device is configured with two VRF groups, vpn-A and vpn-B.

Figure 18: Traffic between MPLS Networks



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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security l3vpn vrf-group vpn-A vrf VRF-A1
set security l3vpn vrf-group vpn-A vrf VRF-A2
set security l3vpn vrf-group vpn-A1 vrf VRF-A11
set security l3vpn vrf-group vpn-A1 vrf VRF-A12
set security l3vpn vrf-group vpn-B vrf VRF-B1
set security l3vpn vrf-group vpn-B vrf VRF-B2
set security l3vpn vrf-group vpn-B1 vrf VRF-B11
set security l3vpn vrf-group vpn-B1 vrf VRF-B12
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match source-address any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match destination-address
    any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match application any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match
    source-l3vpn-vrf-group vpn-A1
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy then permit
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match source-address any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match destination-address
    any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match application any
```

```
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match
   source-l3vpn-vrf-group vpn-B1
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy then permit
```

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 source NAT mapping for a single VRF:

1. In Layer 3 VPNs create a VRF group vpn-A with VRF instances A1and A2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-A vrf VRF-A1
user@host#set l3vpn vrf-group vpn-A vrf VRF-A2
```

2. In Layer 3 VPNs create a VRF group vpn-A1 with VRF instances A11and A12.

```
[edit security]
user@host#set l3vpn vrf-group vpn-A1 vrf VRF-A11
user@host#set l3vpn vrf-group vpn-A1 vrf VRF-A12
```

3. Create another VRF group vpn-B with VRF instances B1 and B2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-B vrf VRF-B1
user@host#set l3vpn vrf-group vpn-B vrf VRF-B2
```

4. Create another VRF group vpn-B1 with VRF instances B11 and B12.

```
[edit security]
user@host#set l3vpn vrf-group vpn-B1 vrf VRF-B11
user@host#set l3vpn vrf-group vpn-B1 vrf VRF-B12
```

5. Create a security policy to permit vpn-A1 traffic.

```
[edit security policies from-zone GRE-1_Zone to-zone GRE-2_Zone]
user@host# set policy vrf-a_policy match source-address any
user@host# set policy vrf-a_policy match destination-address any
```

```
user@host# set policy vrf-a_policy match application any
user@host# set policy vrf-a_policy match destination-l3vpn-vrf-group vpn-A1
user@host# set policy vrf-a_policy then permit
```

6. Create a security policy to permit vpn-B1 traffic.

```
[edit security policies from-zone GRE-1_Zone to-zone GRE-2_Zone]
user@host# set policy vrf-b_policy match source-address any
user@host# set policy vrf-b_policy match destination-address any
user@host# set policy vrf-b_policy match application any
user@host# set policy vrf-b_policy match destination-l3vpn-vrf-group vpn-B1
user@host# set policy vrf-b_policy then permit
```

Results

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

```
[edit]
user@host# show security policies
from-zone GRE-1_Zone to-zone GRE-2_Zone {
    policy vrf-a_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            destination-l3vpn-vrf-group vpn-A1;
        }
        then {
            permit;
        }
    }
    policy vrf-b_policy {
        match {
            source-address any;
            destination-address any;
            application any;
            destination-l3vpn-vrf-group vpn-B1;
        }
        then {
            permit;
```

```
    }
  }
}
```

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

Verification

Verifying Security Policies

Purpose

Verify that configuration output of security policies.

Action

From operational mode, enter the **show security policies** command to display a summary of all the security policies configured on the device.

```
user@root> show security policies
Default policy: permit-all
From zone: GRE-1_Zone, To zone: GRE-2_Zone
  Policy: vrf-a_policy, State: enabled, Index: 4, Scope Policy: 0, Sequence number:
 1
    Source L3VPN VRF Group: any
    destination L3VPN VRF-Group: vpn-A1
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
From zone: GRE-1_Zone, To zone: GRE-2_Zone
  Policy: vrf-b_policy, State: enabled, Index: 5, Scope Policy: 0, Sequence number:
 2
    Source L3VPN VRF Group: any
    destination L3VPN VRF-Group: vpn-B1
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

Example: Configuring a Security Policy to Permit VRF-Based Traffic from MPLS Network to MPLS Network using NAT and VRF Group

IN THIS SECTION

- Requirements | 154
- Overview | 154
- Configuration | 155
- Verification | 160

This example describes how to configure the routing group and permit traffic between MPLS networks using NAT.

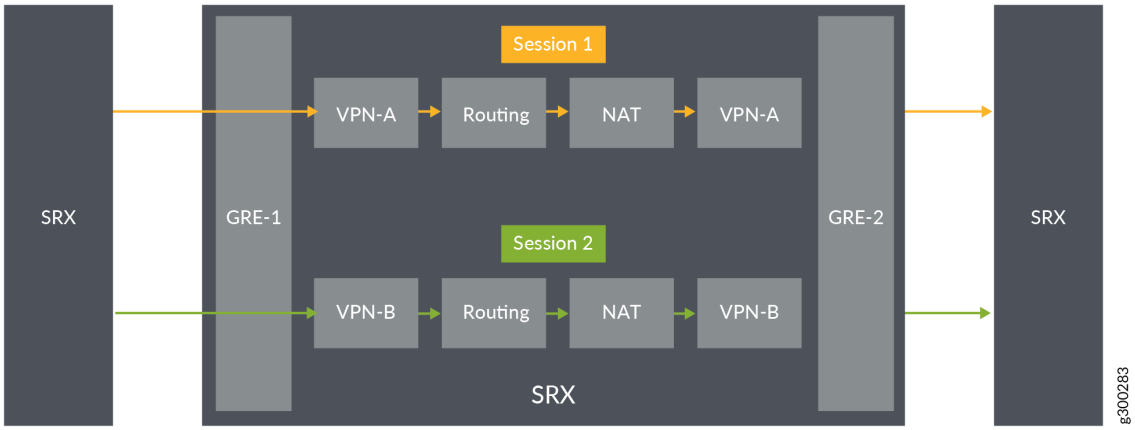
Requirements

- Understand how SRX Series devices work in an SD-WAN deployment for NAT.
- Understand Virtual Routing and Forwarding Instances. See "Virtual Routing and Forwarding Instances in SD-WAN Deployments" on page 117.

Overview

In Figure 19 on page 155, an SRX Series device is configured the routing group and permit traffic between MPLS networks using NAT. The SRX Series device is configured with the VRF groups, vpn-A, vpn-A1, vpn-B, and vpn-B1.

Figure 19: Traffic Permit between MPLS Networks with NAT



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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security l3vpn vrf-group vpn-A vrf VRF-A1
set security l3vpn vrf-group vpn-A vrf VRF-A2
set security l3vpn vrf-group vpn-A1 vrf VRF-A11
set security l3vpn vrf-group vpn-A1 vrf VRF-A12
set security l3vpn vrf-group vpn-B vrf VRF-B1
set security l3vpn vrf-group vpn-B vrf VRF-B2
set security l3vpn vrf-group vpn-B1 vrf VRF-B11
set security l3vpn vrf-group vpn-B1 vrf VRF-B12
set security nat source pool vrf-a_p address 203.0.113.200
set security nat source rule-set vrf-a_rs from routing-group vpn-A
set security nat source rule-set vrf-a_rs to routing-group vpn-A1
set security nat source rule-set vrf-a_rs rule rule1 match source-address 192.168.1.200
set security nat source rule-set vrf-a_rs rule rule1 then source-nat pool vrf-a_p
set security nat source pool vrf-b_p address 203.0.113.201
set security nat source rule-set vrf-b_rs from routing-group vpn-B
set security nat source rule-set vrf-b_rs to routing-group vpn-B1
set security nat source rule-set vrf-b_rs rule rule2 match source-address 192.168.1.201
set security nat source rule-set vrf-b_rs rule rule2 then source-nat pool vrf-b_p
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match source-address any
```

```
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match destination-address
    any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match application any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy match
    source-l3vpn-vrf-group vpn-A1
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-a_policy then permit
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match source-address any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match destination-address
    any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match application any
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy match
    source-l3vpn-vrf-group vpn-B1
set security policies from-zone GRE-1_Zone to-zone GRE-2_Zone policy vrf-b_policy then permit
```

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 source NAT mapping for a single VRF:

1. In Layer 3 VPNs create a VRF group vpn-A with VRF instances A1and A2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-A vrf VRF-A1
user@host#set l3vpn vrf-group vpn-A vrf VRF-A2
```

2. In Layer 3 VPNs create a VRF group vpn-A1 with VRF instances A11and A12.

```
[edit security]
user@host#set l3vpn vrf-group vpn-A1 vrf VRF-A11
user@host#set l3vpn vrf-group vpn-A1 vrf VRF-A12
```

3. Create another VRF group vpn-B with VRF instances B1 and B2.

```
[edit security]
user@host#set l3vpn vrf-group vpn-B vrf VRF-B1
user@host#set l3vpn vrf-group vpn-B vrf VRF-B2
```

4. Create another VRF group vpn-B1 with VRF instances B11 and B12.

```
[edit security]
```

```
user@host#set l3vpn vrf-group vpn-B1 vrf VRF-B11
user@host#set l3vpn vrf-group vpn-B1 vrf VRF-B12
```

5. Specify a source NAT IP address pool.

```
[edit security nat source]
user@host# set pool vrf-a_p address 192.168.1.200
user@host# set pool vrf-b_p address 192.168.1.201
```

6. Create a source NAT rule set.

```
[edit security nat source]
user@host# set rule-set rs from routing-group vpn-A
user@host# set rule-set rs from routing-group vpn-B
user@host# set rule-set rs to routing-group vpn-A1
user@host# set rule-set rs to routing-group vpn-B1
```

7. Configure a rule that matches packets and translates per VRF group network traffic to global IP pool.

```
[edit security nat source]
user@host# set rule-set rs rule vrf-a_rs match destination-address 203.0.113.200
user@host# set rule-set rs rule vrf-a_rs then destination-nat pool vrf-a_p
user@host# set rule-set rs rule vrf-b_rs match destination-address 203.0.113.201
user@host# set rule-set rs rule vrf-b_rs then destination-nat pool vrf-b_p
```

8. Create a security policy to permit vpn-A1 traffic.

```
[edit security policies from-zone GRE-1_Zone to-zone GRE-2_Zone]
user@host# set policy vrf-a_policy match source-address any
user@host# set policy vrf-a_policy match destination-address any
user@host# set policy vrf-a_policy match application any
user@host# set policy vrf-a_policy match destination-l3vpn-vrf-group vpn-A1
user@host# set policy vrf-a_policy then permit
```

9. Create a security policy to permit vpn-B1 traffic.

```
[edit security policies from-zone GRE-1_Zone to-zone GRE-2_Zone]
user@host# set policy vrf-b_policy match source-address any
user@host# set policy vrf-b_policy match destination-address any
```

```
user@host# set policy vrf-b_policy match application any
user@host# set policy vrf-b_policy match destination-l3vpn-vrf-group vpn-B1
user@host# set policy vrf-b_policy then permit
```

Results

From configuration mode, confirm your configuration by entering the **show security nat** and **show security policies** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security nat
   source {
      pool vrf-a_p {
         address {
            203.0.113.200/32;
         }
      }
      pool vrf-b_p {
         address {
            203.0.113.201/32;
         }
      }
      rule-set vrf-a_rs {
         from routing-group vpn-A;
         to routing-group vpn-A1;
         rule rule1 {
            match {
               source-address 192.168.1.200/32;
            }
            then {
               source-nat {
                  pool {
                     vrf-a_p;
                  }
               }
            }
         }
         rule-set vrf-b_rs {
            from routing-group vpn-B;
            to routing-group vpn-B1;
            match {
               source-address 192.168.1.201/32;
```

```
            }
            then {
               source-nat {
                  pool {
                     vrf-b_p;
                  }
               }
            }
         }
      }
   }
```

```
[edit]
user@host# show security policies
from-zone GRE-1_Zone to-zone GRE-2_Zone {
   policy vrf-a_policy {
      match {
         source-address any;
         destination-address any;
         application any;
         destination-l3vpn-vrf-group vpn-A1;
      }
      then {
         permit;
      }
   }
   policy vrf-b_policy {
      match {
         source-address any;
         destination-address any;
         application any;
         destination-l3vpn-vrf-group vpn-B1;
      }
      then {
         permit;
      }
   }
}
```

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

Verification

Verifying Security Policies

Purpose

Verify that there is traffic matching the source NAT rule.

Action

From operational mode, enter the **show security nat source rule all** command. In the Translation hits field, verify whether there is traffic that matches the destination NAT rule.

```
user@host> show security nat source rule all
Total source-nat rules: 2
Total referenced IPv4/IPv6 ip-prefixes: 2/0
Source NAT rule      : vrf-a_r
Rule-set          : rs
  Rule-Id                   : 1
  Rule position             : 1
  From routing-group        : vpn-A
  To zone                   : GE_Zone1
  Source addresses         : 203.0.113.200   - 203.0.113.200
  Action                    : vrf-a_p
  Translation hits          : 0
    Successful sessions     : 0
    Failed sessions         : 0
  Number of sessions        : 0
Source NAT rule      : vrf-b_r
 Rule-set         : rs
  Rule-Id                   : 2
  Rule position             : 2
  From routing-group        : vpn-A
  To zone                   : GE_Zone
  Destination addresses     : 203.0.113.201   - 203.0.113.201
  Action                    : vrf-b_p
  Translation hits          : 0
    Successful sessions     : 0
    Failed sessions         : 0
  Number of sessions        : 0
```

From operational mode, enter the **show security policies** command to display a summary of all the security policies configured on the device.

```
user@root> show security policies
Default policy: permit-all
```

```
From zone: GRE-1_Zone, To zone: GRE-2_Zone
  Policy: vrf-a_policy, State: enabled, Index: 4, Scope Policy: 0, Sequence number:
 1
    Source L3VPN VRF Group: any
    destination L3VPN VRF Group: vpn-A1
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
From zone: GRE-1_Zone, To zone: GRE-2_Zone
  Policy: vrf-b_policy, State: enabled, Index: 5, Scope Policy: 0, Sequence number:
 2
    Source L3VPN VRF Group: any
    destination L3VPN VRF Group: vpn-B1
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: permit
```

RELATED DOCUMENTATION

- *NAT for VRF Routing Instance*
- *Understanding ALG Support for VRF Routing Instance*
- *Configuring Security Policies for a VRF Routing Instance*

3

CHAPTER

Flow-Based Processing for IPv6

IPv6 Flow-Based Processing | **163**

Understanding Path MTU Messages for IPv6 Packets | **178**

IPv6 Packets Header Overview | **180**

IPv6 Flow-Based Processing

IN THIS SECTION

- IPv6 Advanced Flow | 163
- Understanding IPv6 Flow Processing on SRX5400, SRX5600, and SRX5800 devices | 165
- Enabling Flow-Based Processing for IPv6 Traffic | 168
- Flow-Based Processing for IPv6 Traffic on Security Devices | 169
- Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways | 171

This topic covers information on flow processing for IPv6 traffic and IPv6 sessions.

IPv6 Advanced Flow

IN THIS SECTION

- Understanding Sessions for IPv6 Flow | 165

IPv6 advanced flow adds IPv6 support for firewall, NAT, NAT-PT, multicast (local link and transit), IPsec, IDP, JSF framework, TCP Proxy, and Session manager on SRX Series devices. MIBs are not used in the IPv6 flow.

In order to avoid the impact on the current IPv4 environment, IPv6 security is used. If IPv6 security is enabled, extended sessions and gates are allocated. The existing address fields and gates are used to store the index of extended sessions or gates. If IPv6 security is disabled, IPv6 security-related resources are not allocated.

New logs are used for IPv6 flow traffic to prevent impact on performance in the existing IPv4 system.

The behavior and implementation of the IPv6 advanced flow are the same as those of IPv4 in most cases.

The implementations of sessions, gates, ip-actions, processing of multithread, distribution, locking, synchronization, serialization, ordering, packet queuing, asynchronous messaging, IKE traffic issues, sanity check, and queues for IPv6 are similar to IPv4 implementations.

Some of the differences are explained below:

- **Header Parse** IPv6 advanced flow stops parsing the headers and interprets the packet as the corresponding protocol packet if it encounters the following extension headers:
  - TCP/UDP
  - ESP/AH
  - ICMPv6

  IPv6 advanced flow continues parsing headers if it encounters the following extension headers:
  - Hop-by-Hop
  - Routing and Destination, Fragment

  IPv6 advanced flow interprets the packets as an unknown protocol packet if it encounters the extension header **No Next Header**
- **Sanity Checks—** IPv6 advanced flow supports the following sanity checks:
  - TCP length
  - UDP length
  - Hop-by-hop
  - IP data length error
  - Layer 3 sanity checks (for example, IP version and IP length)
  - **ICMPv6 Packets** In IPv6 advanced flow, the ICMPv6 packets share the same behavior as normal IPv6 traffic with the following exceptions:
    - Embedded ICMPv6 packet
    - Path MTU message
- **Host Inbound and Outbound Traffic** IPv6 advanced flow supports all route and management protocols running on the Routing Engine (RE), including OSPF v3, RIPng, Telnet, and SSH. Note that no flow label is used in the flow.
- **Tunnel Traffic** IPv6 advanced flow supports the following tunnel types:
  - IPv4 IP-IP
  - IPv4 GRE

  - IPv4 IPsec
  - Dual-stack lite
- **Events and Logs** The following logs are for IPv6-related flow traffic:
  - RT_FLOW_IPVX_SESSION_DENY
  - RT_FLOW_IPVX_SESSION_CREATE
  - RT_FLOW_IPVX_SESSION_CLOSE

Understanding Sessions for IPv6 Flow

This topic gives an overview of flow-based sessions.

Most packet processing occurs in the context of a flow, including management of policies, zones, and most screens. A session is created for the first packet of a flow for the following purposes:

- To store most of the security measures to be applied to the packets of the flow.
- To cache information about the state of the flow. For example, logging and counting information for a flow is cached in its session. (Also, some stateful firewall screens rely on threshold values that pertain to individual sessions or across all sessions.)
- To allocate resources required for features for the flow.
- To provide a framework for features such as Application Layer Gateways (ALGs).

Understanding IPv6 Flow Processing on SRX5400, SRX5600, and SRX5800 devices

This topic introduces the architecture for the SRX5400, SRX5600, and SRX5800 devices. Flow processing on these devices is similar to that on branch SRX Series devices.

These devices include I/O cards (IOCs) and Services Processing Cards (SPCs) that each contain processing units that process a packet as it traverses the device. These processing units have different responsibilities.

- A Network Processing Unit (NPU) runs on an IOC. An IOC has one or more NPUs. An NPU processes packets discretely and performs basic flow management functions.

  When an IPv6 packet arrives at an IOC, the packet flow process begins.

  - The NPU performs the following IPv6 sanity checks for the packet:
    - For the IPv6 basic header, it performs the following header checks:

- Version. It verifies that the header specifies IPv6 for the version.
- Payload length. It checks the payload length to ensure that the combined length of the IPv6 packet and the Layer 2 header is shorter than the Layer 2 frame length.
- Hop limit. It checks to ensure that the hop limit does not specify 0 (zero).
- Address checks. It checks to ensure that the source IP address does not specify ::0 or FF::00 and that the destination IP address does not specify ::0 or ::1.

- The NPU performs IPv6 extension header checks, including the following:
  - Hop-by-hop options. It verifies that this is the first extension header to follow the IPv6 basic header.
  - Routing extension. It verifies that there is only one routing extension header.
  - Destination options. It verifies that no more than two destination options extension headers are included.
  - Fragment. It verifies that there is only one fragment header.

    NOTE: The NPU treats any other extension header as a Layer 4 header.

- The NPU performs Layer 4 TCP, UDP, and ICMP6 protocol checks, including the following:
  - UDP. It checks to ensure that IP Payload Length packets, other than a first-fragment packet, are at least 8 bytes long.
  - TCP. It checks to ensure that IP Payload Length packets, other than a first-fragment packet, are at least 20 bytes long.
  - ICMPv6. It checks to ensure that IP Payload Length packets, other than a first-fragment packet, are at least 8 bytes long.

- If the packet specifies a TCP or a UDP protocol, the NPU creates a tuple from the packet header data using the following information:
  - Source IP address
  - Destination IP address
  - Source port
  - Destination port
  - Protocol
  - Virtual router identifier (VRID)

    The device looks up the VRID from a VRID table.

- For Internet Control Message Protocol version 6 (ICMPv6) packets, the tuple contains the same information as used for the TCP and the UDP search key, except for the source and destination port fields. The source and destination port fields are replaced with the following information extracted from the ICMPv6 packet:
  - For ICMP error packets: The pattern "0x00010001"
  - For ICMP information packets: The type, or code, field identifier
- For packets with an Authentication Header (AH) or an Encapsulating Security Payload (ESP) header, the search key is the same as that used for the TCP and the UDP tuple, except for the source and destination port fields. In this case, the security parameter index (SPI) field value is used instead of the source and destination ports. For Encapsulating Security Payload (ESP) header and Authentication Header (AH), before enhancements to the cenral point architecture it is hashed by the 3-tuple and the security parameter index (SPI) field, after enhancements to the cenral point architecture it is hashed by an IP pair.
- If a session exists for the packet's flow, the NPU sends the packet to the SPU that manages the session.
- If a matching session does not exist,
  - The NPU sends the packet information to the central point, which creates a pending session.
  - The central point selects an SPU to process the packet and create sessions for it.
  - The SPU then sends session creation messages to the central point and the ingress and egress NPUs, directing them to create a session for the packet flow.

- A central point, which can run on a dedicated SPU, or share the resources of one if there is only one SPU. A central point takes care of arbitration and allocation of resources, and it distributes sessions in an intelligent way. The central point assigns an SPU to be used for a particular session when the SPU processes the first packet of its flow.
  - For SRX5000 line devices, the central point architecture is divided into two modules—the application central point and the distributed central point (DCP). The App-CP is responsible for global resource management and loading balancing, while DCP is responsible for traffic identification (global session matching). The App-CP functionality runs on the dedicated central point SPU, while the DCP functionality is distributed to the rest of the SPUs.
- One or more SPUs that run on a Services Processing Card (SPC). All flow-based services for a packet are executed on a single SPU, within the context of a session that is set up for the packet flow.

  The SPC for SRX5000 line devices has two SPUs.

  Several SPCs can be installed in a chassis.

  Primarily, an SPU performs the following tasks:
  - It manages the session and applies security features and other services to the packet.
  - It applies packet-based stateless firewall filters, classifiers, and traffic shapers.

- If a session does not already exist for a packet, the SPU sends a request message to the NPU that performed the search for the packet's session, to direct it to add a session for it.

These discrete, cooperating parts of the system store the information identifying whether a session exists for a stream of packets and the information against which a packet is matched to determine if it belongs to an existing session.

Enabling Flow-Based Processing for IPv6 Traffic

You have the following options for handling IPv6 traffic:

- Drop—Do not forward IPv6 packets. This is the default behavior.
- Packet-based forwarding—Do not create a session and process according to packet-based features only (includes firewall filters and class of service).
- Flow-based forwarding—Create a session and process according to packet-based features (including firewall filters and class of service) but also flow-based security features, such as screens and firewall security policy.

To enable flow-based processing for IPv6 traffic, modify the **mode** statement at the [**edit security forwarding-options family inet6**] hierarchy level:

```
security {
    forwarding-options {
        family {
            inet6 {
                mode flow-based;
            }
        }
    }
}
```

The following example shows the CLI commands you use to configure forwarding for IPv6 traffic:

```
[edit]
user@host# set security forwarding-options family inet6 mode ?
```

```
Possible completions:
  drop                 Disable forwarding
  flow-based           Enable flow-based forwarding
  packet-based         Enable packet-based forwarding
```

```
[edit]
user@host# set security forwarding-options family inet6 mode flow-based

user@host# show security forwarding-options
```

```
family {
    inet6 {
        mode flow-based;
    }
}
```

If you change the forwarding option mode for IPv6, you might need to perform a reboot to initialize the configuration change. Table 9 on page 169 summarizes device status upon configuration change.

Table 9: Device Status Upon Configuration Change

Configuration Change	Commit Warning	Reboot Required	Impact on Existing Traffic Before Reboot	Impact on New Traffic Before Reboot
Drop to flow-based	Yes	Yes	Dropped	Dropped
Drop to packet-based	No	No	Packet-based	Packet-based
Flow-based to packet-based	Yes	Yes	None	Flow sessions created
Flow-based to drop	Yes	Yes	None	Flow sessions created
Packet-based to flow-based	Yes	Yes	Packet-based	Packet-based
Packet-based to drop	No	No	Dropped	Dropped

Flow-Based Processing for IPv6 Traffic on Security Devices

Flow-based processing mode is required for security features such as zones, screens, and firewall policies to function. By default, the SRX Series device is enabled for flow-based forwarding for IPv6 traffic on all devices, apart from the SRX300 Series and SRX550M devices that are set to drop mode. Starting with Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, for the SRX1500 series, SRX4100, SRX4200, SRX5400, SRX5600, SRX5800 and vSRX devices, you do *not* need to reboot the device when you are switching modes between flow mode, packet mode, and drop mode. For SRX300 Series and SRX550M devices, you *must* reboot the device when switching between flow mode, packet mode, and drop mode.

SRX300 Series and the SRX550M Devices

When IPv6 is configured on SRX300 Series and the SRX550M devices, the default behavior is set to drop mode because of memory constraints. In this case, you must reboot the device after changing the processing mode from the drop mode default to flow-based processing mode or packet-based processing mode—that is, between modes on these devices.

> **NOTE:** For drop mode processing, the traffic is dropped directly, it is not forwarded. It differs from packet-mode processing for which the traffic is handled but no security processes are applied.

To process IPv6 traffic on SRX300 Series and the SRX550M devices, you need to configure IPv6 addresses for the transit interfaces that receive and forward the traffic. For information about the inet6 protocol family and procedures for configuring IPv6 addresses for interfaces.

Configuring an SRX Series Device as a Border Router

When an SRX Series device of any type is enabled for flow-based processing or drop mode, to configure the device as a border router you must change the mode to packet-based processing for MPLS. In this case, to configure the SRX device to packet mode for MPLS, use the **set security forwarding-options family mpls mode packet-based** statement.

> **NOTE:** As mentioned, for SRX300 Series and the SRX550M devices, whenever you change processing modes, you must reboot the device.

Enabling Flow-Based Processing for IPv6 Traffic on SRX300 Series and SRX550M Devices

To enable flow-based forwarding for IPv6 traffic on SRX300 Series and the SRX550M devices, modify the mode at the [**edit security forwarding-options family inet6**] hierarchy level:

```
security {
    forwarding-options {
        family {
            inet6 {
                mode flow-based;
            }
        }
    }
}
```

To configure forwarding for IPv6 traffic on SRX300 Series or an SRX500M device:

1. Change the forwarding option mode for IPv6 to flow-based.

```
[edit]
user@host# security forwarding-options family inet6 mode flow-based
```

2. Review your configuration.

```
[edit]
user@host# show security forwarding-options
  family {
    inet6 {
      mode flow-based;
    }
  }
```

3. Commit the configuration.

```
[edit]
user@host# commit
```

4. Reboot the device.

NOTE: For SRX300 Series and SRX500M devices, the device discards IPv6 type 0 Routing Header (RH0) packets.

Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways

Purpose

You can display flow and session information about one or more sessions with the **show security flow session** command. IPv6 sessions are included in aggregated statistics.

You can use the following filters with the **show security flow session** command: application, destination-port, destination-prefix, family, idp, interface, nat, protocol, resource-manager, session-identifier, source-port, source-prefix, and tunnel.

> **NOTE:** Except for the session-identifier filter, the output of all the other filters can be viewed in brief, summary, and extensive mode. Brief mode is the default mode. The output of the session-identifier filter can be viewed only in the brief mode.

You can use the same filter options with the **clear security flow session** command to terminate sessions.

Action

The following examples show how to use IPv6-related filters to display summaries and details for IPv6 sessions.

> **NOTE:** Starting in Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, many of these session summaries include CP session IDs.

Filtered summary report based on family

`root>` **show security flow session summary family ?**

```
Possible completions:
  inet                 Show IPv4 sessions
  inet6                Show IPv6/IPv6-NATPT sessions
```

`root>` **show security flow session summary family inet6**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2

Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
```

```
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 1
Sessions in other states: 0
Total sessions: 1
```

Filtered detailed report based on family

root> **show security flow session family ?**

```
Possible completions:
  inet                 Show IPv4 sessions
  inet6                Show IPv6/IPv6-NATPT sessions
```

root> **show security flow session family inet6**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000026, Policy name: default-policy-00/2, Timeout: 1794, Valid
  In: 2001:db8::10/64712 -> 2001:db8::4/21;tcp If: ge-7/1/0.0, Pkts: 8, Bytes:
562, CP Session ID: 430000025
  Out: 2001:db8::4/21  --> 2001:db8::10/64712;tcp, If: ge-7/1/1.0, Pkts: 12, Bytes:
 1014, CP Session ID: 430000025
Total sessions: 1
```

Filtered brief report based on family

root> **show security flow session family inet brief**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000031, Policy name: default-policy-00/2, Timeout: 48, Valid
  In: 203.0.113.8/3 --> 198.51.100.11/43053;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes:
 84, CP Session ID: 410000039
  Out: 198.51.100.11/43053 --> 203.0.113.8/3;icmp, If: ge-7/1/1.0, Pkts: 0, Bytes:
 0, CP Session ID: 410000039
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Session ID: 420000034, Policy name: default-policy-00/2, Timeout: 48, Valid
  In: 203.0.113.8/4 --> 198.51.100.11/43053;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes:
 84, CP Session ID: 420000041
  Out: 198.51.100.11/43053  --> 203.0.113.8/4;icmp, If: ge-7/1/1.0, Pkts: 0, Bytes:
```

```
 0, CP Session ID: 420000041
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000042, Policy name: default-policy-00/2, Timeout: 44, Valid
  In: 203.0.113.8/2 --> 198.51.100.11/43053;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes:
 84, CP Session ID: 430000041
  Out: 198.51.100.11/43053 --> 203.0.113.8/2;icmp, If: ge-7/1/1.0, Pkts: 0, Bytes:
 0, CP Session ID: 430000041
Total sessions: 1


2001:dbf8::6:2/32
```

Filtered detailed report based on an IPv6 source-prefix

`root>` **show security flow session source-prefix 2001:dbf8::**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000066, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 2001:dbf8::6:2/3 > 2001:dbf8:5::2/7214;icmp6, If: ge-7/1/0.0, Pkts: 1, Bytes:
 104, CP Session ID: 410000076
  Out: 2001:dbf8:5::2/7214 --> 2001:dbf8:5::2/323;icmp6, If: .local..0, Pkts: 1,
Bytes: 104, CP Session ID: 410000076

Session ID: 410000068, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 2001:dbf8::6:2/4 --> 2001:dbf8:5::2/7214;icmp6, If: ge-7/1/0.0, Pkts: 1,
Bytes: 104, CP Session ID: 410000077
  Out: 2001:dbf8:5::2/7214 --> 2001:dbf8::6:2/4;icmp6, If: .local..0, Pkts: 1,
Bytes: 104, CP Session ID: 410000077
Total sessions: 2

Flow Sessions on FPC10 PIC2:

Session ID: 420000067, Policy name: default-policy-00/2, Timeout: 28, Valid
  In: 2001:dbf8::6:2/4 --> 2001:dbf8:5::3/6702;icmp6, If: ge-7/1/0.0, Pkts: 1,
Bytes: 104, CP Session ID: 420000080
  Out: 2001:dbf8:5::3/6702 --> 2001:dbf8::6:2/4 ;icmp6, If: ge-7/1/1.0, Pkts: 0,
Bytes: 0, CP Session ID: 420000080
Total sessions: 1

Flow Sessions on FPC10 PIC3:
```

```
Session ID: 430000077, Policy name: default-policy-00/2, Timeout: 28, Valid
  In: 2001:dbf8::6:2/3 --> 2001:dbf8:5::3/6702;icmp6, If: ge-7/1/0.0, Pkts: 1,
Bytes: 104, CP Session ID: 430000075
  Out: 2001:dbf8:5::3/6702 --> 2001:dbf8::6:2/3;icmp6, If: ge-7/1/1.0, Pkts: 0,
Bytes: 0, CP Session ID: 430000075

Session ID: 430000078, Policy name: default-policy-00/2, Timeout: 30, Valid
  In: 2001:dbf8::6:2/5 --> 2001:dbf8:5::3/6702, If: ge-7/1/0.0, Pkts: 1, Bytes:
104, CP Session ID: 430000076
  Out: 2001:dbf8:5::3/6702 --> 2001:dbf8::6:2/5;icmp6, If: ge-7/1/1.0, Pkts: 0,
Bytes: 0, CP Session ID: 430000076

Session ID: 430000079, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 2001:dbf8::6:2/5 --> 2001:dbf8:5::1/7214;icmp6, If: ge-7/1/0.0, Pkts: 1,
Bytes: 104, CP Session ID: 430000077
  Out: 2001:dbf8:5::1/7214 --> 2001:dbf8::6:2/5;icmp6, If: .local..0, Pkts: 1,
Bytes: 104, CP Session ID: 430000077
Total sessions: 3
```

Multiple-filtered detailed report based on family, protocol and source-prefix

`root>` **show security flow session family inet protocol icmp source-prefix 2001:db8::**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000074, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 2001:dbf8::6:2/1 --> 2001:dbf8:8::2/26935;icmp, If: ge-7/1/0.0, Pkts: 1,
Bytes: 84, CP Session ID: 410000195
  Out: 2001:dbf8:8::2 --> 2001:dbf8::6:2/1;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes:
84, CP Session ID: 410000195
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Session ID: 420000075, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 2001:dbf8::6:2/3 --> 2001:dbf8::6:2/26935;icmp, If: ge-7/1/0.0, Pkts: 1,
Bytes: 84, CP Session ID: 420000159
  Out: 2001:dbf8::6:2/26935 --> 2001:dbf8::6:2/3;icmp, If: ge-7/1/1.0, Pkts: 1,
Bytes: 84, CP Session ID: 420000159
Total sessions: 1

Flow Sessions on FPC10 PIC3:
```

```
Session ID: 430000085, Policy name: default-policy-00/2, Timeout: 2, Valid
  In:  2001:dbf8::6:2/4 --> 2001:dbf8::6:2/26935;icmp, If: ge-7/1/0.0, Pkts: 1,
Bytes: 84, CP Session ID: 430000083
  Out: 2001:dbf8::6:2/26935 -->  2001:dbf8::6:2/4;icmp, If: ge-7/1/1.0, Pkts: 1,
Bytes: 84, CP Session ID: 430000083
Total sessions: 1
```

Clearing all sessions, including IPv6 sessions

`root>` **clear security flow session all**

```
This command may terminate the current session too.
Continue? [yes,no] (no) yes
```

```
0 active sessions cleared
1 active sessions cleared
1 active sessions cleared
1 active sessions cleared
```

Clearing only IPv6 sessions

`root>` **clear security flow session family ?**

```
Possible completions:
  inet                 Clear IPv4 sessions
  inet6                Clear IPv6/IPv6-NATPT sessions
```

`root>` **clear security flow session family inet6**

```
0 active sessions cleared
1 active sessions cleared
1 active sessions cleared
1 active sessions cleared
```

Release History Table

Release	Description
15.1X49-D70	By default, the SRX Series device is enabled for flow-based forwarding for IPv6 traffic on all devices, apart from the SRX300 Series and SRX550M devices that are set to drop mode. Starting with Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, for the SRX1500 series, SRX4100, SRX4200, SRX5400, SRX5600, SRX5800 and vSRX devices, you do *not* need to reboot the device when you are switching modes between flow mode, packet mode, and drop mode. For SRX300 Series and SRX550M devices, you *must* reboot the device when switching between flow mode, packet mode, and drop mode.
15.1X49-D30	Starting in Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, many of these session summaries include CP session IDs.

RELATED DOCUMENTATION

IPv6 Packets Header Overview | 180

Understanding Path MTU Messages for IPv6 Packets

This topics describes path maximum transmission unit (MTU) and explains how the flow module for SRX Series devices processes and uses path MTU messages.

Every link has an MTU size that specifies the size of the largest packet the link can transmit. A larger MTU size means that fewer packets are required to transmit a certain amount of data. To achieve the best data transmission performance, IPv6 data packets sent from one node (the source) to another node (the destination) should be the largest possible size that can traverse the path between the nodes. (Larger and fewer packets constrain the cost of packet header processing and routing processes that can affect transmission performance.)

However, for a packet to successfully traverse the path from the source node to the destination node, the MTU size of the source node interface must be no larger than that of the smallest MTU size of all nodes on the path between the source and destination. This value is referred to as the path maximum transmission unit (path MTU). If a packet is larger than a link's MTU size, it is likely that the link will drop it. For IPv6, an intermediate node cannot fragment a packet.

IPv6 defines a standard mechanism called path MTU discovery that a source node can use to learn the path MTU of a path that a packet is likely to traverse. If any of the packets sent on that path are too large to be forwarded by a node along the path, that node discards the packet and returns an ICMPv6 Packet

Too Big message. The source node can then adjust the MTU size to be smaller than that of the node that dropped it and sent the ICMPv6 message, and then retransmit the packet. A source node might receive Packet Too Big messages repeatedly until its packet traverses all nodes along the path successfully.

> **NOTE:** On all SRX Series devices, the Routing Engine cannot detect the path MTU of an IPv6 multicast address (with a large size packet).

After the path MTU size is determined and the appropriate MTU size is set, an outgoing packet might be routed along a different path with a node whose link MTU size is smaller than the path MTU size determined previously. In this case, the flow module engages the path MTU discovery process again.

When the flow module receives an ICMP Packet Too Big message with a destination address that belongs to it, it:

- Checks to determine if the embedded 5-tuple data of the packet is for a tunnel interface. (That is, it checks to determine if the embedded 5-tuple data matches a tunnel session.) If there is a match, the flow module updates the tunnel interface's MTU size. Then it performs post-fragment processing for the encrypted packets that follow the first packet. Afterward, the flow module delivers the packet to the ICMPv6 stack on the Routing Engine (RE) for it to continue processing it.
- If the packet is a transit one, the flow module searches for a session that matches the packet's embedded 5-tuple data. It is finds a matching session, it delivers the packet to it. If there is no matching session, it drops the packet.

When the flow module receives a packet, before it transmits it to the egress interface, it checks to determine if the MTU size of the egress interface is greater than the packet length.

- If the MTU size is greater than the packet length, it continues to process the packet.
- If the MTU size is less than the packet length, it drops the packet and sends an ICMPv6 Packet Too Big message to the source node.

> **NOTE:** When chassis cluster is configured and the path MTU updates the MTU of the tunnel interface, the flow module does not synchronize the new MTU to peer nodes. The MTU size might be updated again by a larger packet on a peer node, which has no impact on packet transmission.

IPv6 Packets Header Overview

IN THIS SECTION

- The IPv6 Packet Header and SRX Series Overview | 180
- Understanding IPv6 Packet Header Extensions | 180
- Understanding How SRX Series Devices Handle ICMPv6 Packets | 182

This topic identifies the IP version 6 (IPv6) packet header and its extensions and options.

The IPv6 Packet Header and SRX Series Overview

Every IPv6 packet at a minimum has a basic packet header, 40 bytes (320 bits) long. They optionally may have extension headers.

For IPv6 packets, flow processing parses the extension headers and transport layer headers in the following way:

- If the software encounters a TCP, a UDP, an ESP, an AH, or an ICMPv6 header, it parses the header and assumes that the packet payload corresponds to the specified protocol type.
- If the software encounters a hop-by-hop header, a routing and destination header, or a fragment header, it continues to parse the next extension header.
- If it encounters the no-next-header extension header, the software detects that the packet is that of an unknown protocol (protocol equals 0).
- For other extension headers, the software parses the header and identifies the packet as belonging to the protocol indicated by the extension header.

Understanding IPv6 Packet Header Extensions

IPv6 extension headers contain supplementary information used by network devices (such as routers, switches, and endpoint hosts) to decide how to direct or process an IPv6 packet. The length of each extension header is an integer multiple of 8 octets. This allows subsequent extension headers to use 8-octet structures.

Any header followed by an extension header contains a Next Header value that identifies the extension header type. Extension headers always follow the basic IPv6 header in order as shown in Table 10 on page 181:

> **NOTE:** The destination IP address can appear twice, once after the hop-by-hop header and again after the last extension header.

Table 10: IPv6 Extension Headers

Header Name	Purpose
Hop-by-Hop Options	Specifies delivery parameters at each hop on the path to the destination host. **NOTE:** A hop-by-hop option can appear only following the IPv6 basic header. If it is used, it should be the first extension header. It cannot appear after another extension header.
Destination Options	Specifies packet delivery parameters for either intermediate destination devices or the final destination host. When a packet uses this header, the Next Header value of the previous header must be 60.
Routing	Defines strict source routing and loose source routing for the packet. (With strict source routing, each intermediate destination device must be a single hop away. With loose source routing, intermediate destination devices can be one or more hops away.) When a packet uses this header, the Next Header value of the previous header must be 43.
Fragment	Specifies how to perform IPv6 fragmentation and reassembly services. When a packet uses this header, the Next Header value of the previous header must be 44. A source node uses the fragment extension header to tell the destination node the size of the packet that was fragmented so that the destination node can reassemble the packet.
Authentication	Provides authentication, data integrity, and anti-replay protection. When a packet uses this header, the Next Header value of the previous header must be 51.
Encapsulating Security Payload	Provides data confidentiality, data authentication, and anti-replay protection for Encapsulated Security Payload (ESP) packets. When a packet uses this header, the Next Header value of the previous header must be 50.

Table 10: IPv6 Extension Headers *(continued)*

Header Name	Purpose
Destination IP Address	Identifies the host device, or interface on a node, to which the IPv6 packet is to be sent. **NOTE:** The destination address may appear twice, the first instance after the hop limit following the source IP address and the second instance after the final extension header.

Understanding How SRX Series Devices Handle ICMPv6 Packets

This topic explains Internet Control Message Protocol (ICMP), ICMP messages, and how Junos OS for SRX Series Services Gateways uses them.

ICMP provides a framework for reporting packet processing errors, for diagnostic purposes, and for implementation-specific functions. ICMP error messages make it possible for one node to inform another node that something has gone wrong during the course of data transfer. When IP version 6 (IPv6) was defined, the differences between IP version 4 (IPv4) and it were significant enough to require a new version of ICMP.

Every ICMPv6 message is preceded by an IPv6 header and zero or more IPv6 extension headers. The ICMPv6 header is identified by a Next Header value of 58 in the immediately preceding header. This is different from the value used to identify ICMP for IPv4. All ICMPv6 error messages have 32 bits of type-specific data to help the packet recipient locate the embedded invoking packet.

Most ICMPv6 packets have the same characteristics and behavior as normal IPv6 packets, and the Junos OS flow module processes them through first path and fast-path processing in the same way that it does normal IPv6 packets. Table 11 on page 183 shows the ICMPv6 embedded packet types that the flow module handles differently from normal ICMPv6 packets.

For these packets, the flow module uses a tuple that it creates from the embedded ICMPv6 packet to search for a matching session. It continues to process the packet without modifying the maximum transmission unit (MTU) until it finds a matching session, unless it receives an ICMPv6 Packet Too Big message for the interface. In this case, it modifies the MTU size for that interface. If the flow module does not find a matching session or if it cannot obtain a valid IPv6 header from the embedded payload, it drops the packet.

> **NOTE:** A Packet Too Big message is the only kind of ICMPv6 packet that will cause the flow module to modify an interface.

Table 11: ICMPv6 Packets That Junos OS Handles Differently from Other ICMPv6 Packets

Message	Meaning
01-Destination Unreachable	When a packet cannot be delivered because of a problem with the way it is being sent, it is useful to have a feedback mechanism that can tell the source about the problem, including the reason why delivery of the packet failed. For IPv6, the Destination Unreachable message serves this purpose. Each message includes a code that indicates the nature of the problem that caused the packet delivery to fail. It also includes all or part of the packet that could not be delivered, to help the source device resolve the problem. When the flow module encounters a Destination Unreachable ICMP packet whose embedded packet header data matches the 5-tuple data for a session, the software terminates the session.
02-Packet Too Big	When the flow module receives an ICMPv6 Packet Too Big message intended for it, the flow module sends the packet to the ICMP protocol stack on the Routing Engine to engage the path maximum transmission unit (path MTU) discovery process. If the Packet Too Big message does not pertain to the device but rather is a transit packet, the device attempts to match the embedded 5-tuple data with a session. • If a matching session exists, the device delivers it to the source node. • If a matching session does not exist, the device drops the packet **NOTE:** A Packet Too Big message is the only kind of ICMPv6 packet that will cause the flow module to modify an interface.
03-Time Exceeded	When the flow module receives a packet that cannot be delivered because it has exceeded the hop count specified in the basic header hop-by-hop field, it sends this message to inform the packet's source node that the packet was discarded for this reason.

Table 11: ICMPv6 Packets That Junos OS Handles Differently from Other ICMPv6 Packets *(continued)*

Message	Meaning
04-Parameter Problem	When the device finds a problem with a field in the IPv6 header or extension headers that makes it impossible for it to process the packet, the software discards it and sends this ICMPv6 message to the packet's source node, indicating the type and location of the problem.

4

CHAPTER

Monitoring Flow-Based Sessions and Establishing Parameters for Error Handling

Monitoring Security Flow Sessions | **186**

Monitoring X2 Traffic | **200**

Monitoring Security Flow Sessions

IN THIS SECTION

- Monitoring Security Flow Sessions Overview | 186
- Understanding How to Obtain Session Information for SRX Series Services Gateways | 187
- Displaying Global Session Parameters for All SRX Series Services Gateways | 189
- Displaying a Summary of Sessions for SRX Series Services Gateways | 189
- Displaying Session and Flow Information About Sessions for SRX Series Services Gateways | 190
- Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways | 190
- Using Filters to Display Session and Flow Information for SRX Series Services Gateways | 191
- Information Provided in Session Log Entries for SRX Series Services Gateways | 191
- Error Handling Extensions | 196

This topic covers information for monitoring, displaying and verifying of flow sessions using operational mode commands. Thus, you can debug without having to commit or modify your running configuration.

Monitoring Security Flow Sessions Overview

Junos OS allows you to configure and start the monitoring of flow sessions using operational mode commands. Thus, you can debug without having to commit or modify your running configuration. This approach can be especially useful when you do not want to change the state of your device by committing the configuration to turn on trace options.

To configure flow session monitoring, you must define flow filters, specify the output file, and start monitoring. Flow session monitoring does not start unless a filter (at least one) and an output file are specified. Also, defining the filters themselves does not trigger monitoring. You have to explicitly use the **monitor security flow start** and **monitor security flow stop** commands to enable and disable monitoring, respectively.

- Define flow filters—Define the flow sessions that you want to monitor using combinations of match criteria, such as source address, destination address, source port, destination port, IP protocol number, name of the incoming or outgoing interface, and the logical system name. You can delete filters using the **clear monitor security flow filter** command.

  > **NOTE:** Unlike filters defined in the configuration mode, filters defined using operational mode commands are cleared when you reboot your system.

- Specify the output file—Create an output file in which the security flow monitoring information is to be saved. This file is saved in the **/var/log/** directory. You can view the contents of this file by using the **show log *filename*** command. Use the **monitor security flow file** command to specify output file characteristics, such as its maximum size, maximum number, and type.
- Start monitoring—Use the **monitor security flow start** command to start monitoring. Once monitoring starts, any traffic that matches the filters is saved in the specified output file in the **/var/log/** directory. The basic-datapath flag is the default flag and turns on as monitoring starts.

  Use the **monitor security flow stop** command to stop monitoring. Once monitoring stops, the basic-datapath flag is cleared.
- Display monitoring flow information—Use the **show monitoring security flow** command to display details about the monitoring operation.

  > **NOTE:** You can configure flow session monitoring and debugging by using the monitoring operational mode commands and flow traceoptions configuration statements. These two operations cannot run in parallel. When you turn on security flow monitoring, the flow traceoption session is blocked and when the flow traceoption session is running, monitoring of the flow session is blocked.

Understanding How to Obtain Session Information for SRX Series Services Gateways

You can obtain information about the sessions and packet flows active on your device, including detailed information about specific sessions. (The SRX Series device also displays information about failed sessions.) You can display this information to observe activity and for debugging purposes. For example, you can use the show security flow session command:

- To display a list of incoming and outgoing IP flows, including services

- To show the security attributes associated with a flow, for example, the policies that apply to traffic belonging to that flow
- To display the session timeout value, when the session became active, for how long it has been active, and if there is active traffic on the session

> **NOTE:** If an interface NAT is configured and sessions are set up with the NAT using that interface IP address, whenever the interface IP address changes, the sessions set up with NAT get refreshed and new sessions will be setup with new IP address. This you can verify using **show security flow session** CLI command.

Session information can also be logged if a related policy configuration includes the logging option. For the flow session log on all SRX Series devices, policy configuration has been enhanced. Information on the packet incoming interface parameter in the session log for session-init and session-close and when a session is denied by a policy or by the application firewall is provided to meet Common Criteria (CC) Medium Robustness Protection Profiles (MRPP) compliance:

Policy configuration—To configure the policy for the session for which you want to log matches as log **session-init** or **session-close** and to record sessions in syslog:

- **set security policies from-zone untrustZone to-zone trust zone policy policy13 match source-address extHost1**
- **set security policies from-zone untrustZone to-zone trust zone policy policy13 match source-address extHost1**
- **set security policies from-zone untrustZone to-zone trustZone policy policy13 match application junos-ping**
- **set security policies from-zone untrustZone to-zone trustZone policy policy13 then permit**
- **set security policies from-zone untrustZone to-zone trustZone policy policy13 then log session-init**
- **set security policies from-zone untrustZone to-zone trustZone policy policy13 then log session-close**

Example : Flow match policy13 will record the following information in the log:

<14>1 2010-09-30T14:55:04.323+08:00 mrpp-srx550-dut01 RT_FLOW - RT_FLOW_SESSION_CREATE [junos@2626.192.0.2.1.40 source-address="192.0.2.1" source-port="1" destination-address="198.51.100.12" destination-port="46384" service-name="icmp" nat-source-address="192.0.2.1" nat-source-port="1" nat-destination-address="198.51.100.12" nat-destination-port="46384" src-nat-rule-name="None" dst-nat-rule-name="None" protocol-id="1" policy-name="policy1" source-zone-name="trustZone" destination-zone-name="untrustZone" session-id-32="41" packet-incoming-interface="ge-0/0/1.0"] session created 192.0.2.1/1-->198.51.100.12/46384 icmp 192.0.2.1/1-->198.51.100.12/46384 None None 1 policy1 trustZone untrustZone 41 ge-0/0/1.0

<14>1 2010-09-30T14:55:07.188+08:00 mrpp-srx550-dut01 RT_FLOW - RT_FLOW_SESSION_CLOSE [junos@2626.192.0.2.1.40 reason="response received" source-address="192.0.2.1" source-port="1" destination-address="198.51.100.12" destination-port="46384" service-name="icmp" nat-source-address="192.0.2.1" nat-source-port="1" nat-destination-address=”198.51.100.12" nat-destination-port="46384" src-nat-rule-name="None" dst-nat-rule-name="None" protocol-id="1" policy-name="policy1" source-zone-name="trustZone" destination-zone-name="untrustZone" session-id-32="41" packets-from-client="1" bytes-from-client="84" packets-from-server="1" bytes-from-server="84" elapsed-time="0" packet-incoming-interface="ge-0/0/1.0"] session closed response received: 192.0.2.1/1-->198.51.100.12/46384 icmp 192.0.2.1/1-->198.51.100.12/46384 None None 1 policy1 trustZone untrustZone 41 1(84) 1(84) 0 ge-0/0/1.0

Displaying Global Session Parameters for All SRX Series Services Gateways

Purpose

Obtain information about configured parameters that apply to all flows or sessions.

Action

To view session information in the CLI, enter the following command:

```
user@host# show security flow
```

Meaning

The **show security flow** configuration command displays the following information:

- **allow-dns-reply**—Identifies if unmatched incoming Domain Name System (DNS) reply packets are allowed.
- **route-change-timeout**—If enabled, displays the session timeout value to be used on a route change to a nonexistent route.
- **tcp-mss**—Shows the current configuration for the TCP maximum segment size value to be used for all TCP packets for network traffic.
- **tcp-session**—Displays all configured parameters that control session parameters.
- **syn-flood-protection-mode**—Displays the SYN Proxy mode.

Displaying a Summary of Sessions for SRX Series Services Gateways

Purpose

Determine the kinds of sessions on your device, how many of each kind there are—for example, the number of unicast sessions and multicast sessions—the number of failed sessions, the number of sessions that are

currently used and the maximum number of sessions that the device supports. This command also displays the details of the sessions that are currently used. For example, valid sessions, pending sessions, invalidated sessions and sessions in other states.

Action

To view session summary information in the CLI, enter the following CLI command:

```
user@host> show security flow session summary
```

Displaying Session and Flow Information About Sessions for SRX Series Services Gateways

Purpose

Display information about all sessions on your device, including the session ID, the virtual system the session belongs to, the Network Address Translation (NAT) source pool (if source NAT is used), the configured timeout value for the session and its standard timeout, and the session start time and how long the session has been active. The display also shows all standard flow information, including the direction of the flow, the source address and port, the destination address and port, the IP protocol, and the interface used for the session.

Action

To view session flow information in the CLI, enter the following command:

```
user@host> show security flow session
```

Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways

Purpose

When you know the session identifier, you can display all session and flow information for a specific session rather than for all sessions.

Action

To view information about a specific session in the CLI, enter the following command:

```
user@host> show security flow session session-identifier 40000381
```

Using Filters to Display Session and Flow Information for SRX Series Services Gateways

Purpose

You can display flow and session information about one or more sessions by specifying a filter as an argument to the **show security flow session** command. You can use the following filters: application, destination-port, destination-prefix, family, idp, interface, nat, protocol, resource-manager, session-identifier, source-port, source-prefix and tunnel. The device displays the information for each session followed by a line specifying the number of sessions reported on. Here is an example of the command using the source-prefix filter.

Action

To view information about selected sessions using filters in the CLI, enter the following command:

```
user@host> show security flow session source-prefix 10/8
```

Information Provided in Session Log Entries for SRX Series Services Gateways

Session log entries are tied to policy configuration. Each main session event—create, close, and deny—will create a log entry if the controlling policy has enabled logging.

Different fields are logged for session create, session close, and session deny events as shown in Table 12 on page 191, Table 13 on page 192, and Table 14 on page 196. The same field name under each type indicates that the same information is logged, but each table is a full list of all data recorded for that type of session log.

The following table defines the fields displayed in session log entries.

Table 12: Session Create Log Fields

Field	Description
source-address	Source IP address of the packet that created the session.
source-port	Source port of the packet that created the session.
destination-address	Destination IP address of the packet that created the session.
destination-port	Destination port of the packet that created the session.

Table 12: Session Create Log Fields *(continued)*

Field	Description
service-name	Application that the packet traversed (for example, “junos-telnet” for Telnet traffic during the session allowed by a policy that permits native Telnet).
nat-source-address	The translated NAT source address if NAT was applied; otherwise, the source address as above.
nat-source-port	The translated NAT source port if NAT was applied; otherwise, the source port as above.
nat-destination-address	The translated NAT destination address if NAT was applied; otherwise, the destination address as above.
nat-destination-port	The translated NAT destination port if NAT was applied; otherwise, the destination port as above.
src-nat-rule-name	The source NAT rule that was applied to the session (if any). If static NAT is also configured and applied to the session and if source address translation takes place, then this field shows the static NAT rule name.*
dst-nat-rule-name	The destination NAT rule that was applied to the session (if any). If static NAT is also configured and applied to the session and if destination address translation takes place, then this field shows the static NAT rule name.*
protocol-id	The protocol ID of the packet that created the session.
policy-name	The name of the policy that permitted the session creation.
session-id-32	The 32-bit session ID.

** Note that some sessions might have both destination and source NAT applied and the information logged.*

Starting with Junos OS Release 12.1X47-D20 and Junos OS Release 17.3R1, the system log includes information about NAT rule type. Two new src-nat-rule-type and dst-nat-rule-type fileds are introduced in the NAT rule session.

Table 13: Session Close Log Fields

Field	Description
reason	The reason the session was closed.
source-address	Source IP address of the packet that created the session.

Table 13: Session Close Log Fields *(continued)*

Field	Description
source-port	Source port of the packet that created the session.
destination-address	Destination IP address of the packet that created the session.
destination-port	Destination port of the packet that created the session.
service-name	Application that the packet traversed (for example, “junos-telnet” for Telnet traffic during the session allowed by a policy that permits native Telnet).
nat-source-address	The translated NAT source address if NAT was applied; otherwise, the source address as above.
nat-source-port	The translated NAT source port if NAT was applied; otherwise, the source port as above.
nat-destination-address	The translated NAT destination address if NAT was applied; otherwise, the destination address as above.
nat-destination-port	The translated NAT destination port if NAT was applied; otherwise, the destination port as above.
src-nat-rule-name	The source NAT rule that was applied to the session (if any). If static NAT is also configured and applied to the session and if source address translation takes place, then this field shows the static NAT rule name.*
dst-nat-rule-name	The destination NAT rule that was applied to the session (if any). If static NAT is also configured and applied to the session and if destination address translation takes place, then this field shows the static NAT rule name.*
protocol-id	The protocol ID of the packet that created the session.
policy-name	The name of the policy that permitted the session creation.
session-id-32	The 32-bit session ID.
packets-from-client	The number of packets sent by the client related to this session.

Table 13: Session Close Log Fields *(continued)*

Field	Description
bytes-from-client	The number of data bytes sent by the client related to this session.
packets-from-server	The number of packets sent by the server related to this session.
bytes-from-server	The number of data bytes sent by the server related to this session.
elapsed-time	The total session elapsed time from permit to close, given in seconds.
unset	During the session creation, you can set the session close reason as **unset**. The session closes with the reason **unset** if the session installation on the control point is not successful. The reason for session installation varies, for example, nonavailability of memory for nonmanagement session installation.
TCP CLIENT RST	The session was closed by a TCP reset packet sent to it from the client.
TCP SERVER RST	The session was closed by a TCP reset packet sent to it from the server.
TCP FIN	FIN received from either end.
response received	Response received for a packet request (for example, ICMP req-reply).
ICMP error	ICMP error received.
aged out	Session aged out was reached.
ALG	ALG errors closed the session (for example, remote access server (RAS) maximum limit reached).
HA	HA message closed the session.

Table 13: Session Close Log Fields *(continued)*

Field	Description
idle Timeout	There was no traffic for the session before the configured age-out time was reached.
auth	Authentication failed.
IDP	IDP closed the session because of security module (SM) internal error.
synproxy failure	SYN proxy failure closed the session.
synproxy limit	Reason for failure in allocating minor session, need to free original session.
parent closed	Parent session closed.
CLI	Session cleared by a CLI .
CP NACK	CP NACK response received.
CP delete	CP ACK deletion closed the session.
policy delete	Corresponding policy marked for deletion.
fwd session	Session closed because of forwarding session deletion.
multicast route change	Session closed because multicast route changed.
first path reroute, session recreated	The first path is rerouted and session is re-created.
source NAT allocation failure	SPU received ACK message from the central point but failed to receive the DIP resource. Therefore this packet is dropped and the session is closed.
other	Session closed because of all other reasons (for example, the pim reg tun needed refreshing).
error create IKE pass-through template	IKE pass-through template creation errors.
IKE pass-through child session ageout	Session is deleted because the IKE pass through template session has no child.

Table 13: Session Close Log Fields *(continued)*

Field	Description
sess timeout on pending state	Pending session closed because time out timer reached the pending state.
unknown	Session closed because of unknown reasons.

** Note that some sessions might have both destination and source NAT applied and the information logged.*

Table 14: Session Deny Log Fields

Field	Description
source-address	Source IP address of the packet that attempted to create the session.
source-port	Source port of the packet that attempted to create the session.
destination-address	Destination IP address of the packet that attempted to create the session.
destination-port	Destination port of the packet that attempted to create the session.
service-name	Application that the packet attempted to traverse.
protocol-id	The protocol ID of the packet that attempted to create the session.
icmp-type	The ICMP type if the denied packet was ICMP configured; otherwise, this field will be 0.
policy-name	The name of the policy that denied the session creation.

Error Handling Extensions

IN THIS SECTION

- Understanding Chassis Manager FPC Fault Detection and Error Handling Enhancements | **197**

Understanding Chassis Manager FPC Fault Detection and Error Handling Enhancements

IN THIS SECTION

- Error Handling on IOCs and SPCs | **197**
- Error Detection and Management | **197**
- Error Detection Processes | **198**
- Integration with Chassis Cluster | **199**
- Wedge Detection, Reporting, and Management | **199**

The Junos OS Routing Engine and microkernel error detection and management feature on the SRX5400, SRX5600, and SRX5800 devices enables the Routing Engine and the ukernel to accumulate and store the history of all reported error activity and counters for various severity levels. You can configure how errors are handled and specify the severity levels and the actions to perform when an error is detected and a threshold is reached. You can generate and display reports for encountered errors based on stored information.

Starting with Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, error detection enhancements are provided that detect additional errors on IOCs and SPCs and provide enhanced error management. This implementation extends the error detection and management covered in the **show chassis fpc error** topic.

> **NOTE:** This feature is not supported on Routing Engine version 1.

Error Handling on IOCs and SPCs

Starting with Junos OS Release 15.1-X49-D50 and Junos OS Release 17.3R1, the error management enhancements are supported on IOC2 and IOC3 I/O cards (IOCs) and SPC2 Services Processing Cards (SPCs). Some enhancement functions are particular to either the IOC2 and the IOC3 or the SPC2 FPCs, and the differences are called out in this topic.

Error Detection and Management

Error management entails:

- Detecting an error.

  Junos OS monitors the chassis component state to detect a set of error conditions. A detected error can belong to one of the preconfigured error severity levels:

  - Fatal

  - Major
  - Minor
- Identifying the action to take.

  When an error occurs, the system identifies the action to take based on the severity level of the error and the thresholds set and met.

  An FPC maintains a set of error counters for each error severity level. An error counter set consists of a counter that is cumulative across all errors and counters for individual errors and types. It is this information that is stored in the Routing Engine. Each occurrence counter is associated with an error occurrence threshold. There are two threshold levels: one based on the type and the other on severity.
- Executing the action.

  For these enhancements, the preconfigured actions that you can direct the device to take when the Routing Engine's error occurrence count for a given security level reaches the configured threshold are:

  - Reset
  - Offline
  - Alarm
  - Get-state
  - Log

CAUTION: Take care when setting the fault handling actions for SPC2 cards on the SRX5000 line of devices. Consider that if you set the fault handling action on an SPC2 card to offline or reset, when the card is either taken offline or the reboot occurs, the chassis daemon (chassisd) will reboot all of its FPC cards, both SPCs and IOCs—that is, the entire chassis will be rebooted.

Error Detection Processes

With these enhancements, the following error detection processes are enabled and supported:

- Error management on the Routing Engine version 2.
- Error management on ukernel modules on SPC2 cards.
- Error management on the IOC2 and IOC3 cards.
- Driver checks for datapath error detection of wedge conditions.

> **NOTE:** Wedge condition detection for the Trinity Offload Engine driver is supported only on SPC2 cards. That is, it is not supported on the IOC2 and IOC3 cards.

- Wedge detection for host loopback.

> **NOTE:** Wedge condition detection for host loopback is supported only on SPC2 cards. That is, it is not supported on the IOC2 and IOC3 cards.

- Chassis Manager fabric error detection.
- Control path error detections on IOC2 and IOC3 cards.

Integration with Chassis Cluster

In a chassis cluster environment, when an alarm is raised for the first time because of a major or a fatal error, a Redundancy Group 1 (RG1) switchover is triggered. This is the standard behavior on SRX Series devices, and it remains unchanged. However, with these enhancements, the alarm is added to the default fault handling action list for a fatal error. Adding an alarm to the default fault handling list allows the chassis alarm to trigger the RG1 switchover as soon as the fatal error is detected.

Wedge Detection, Reporting, and Management

A wedge condition is caused by an error that blocks network traffic.

This feature detects several types of wedge conditions. It:

- Determines if the wedge is transient or irreversible.
- Records the wedge conditions in statistics and syslogs.
- Alerts network administrators to irreversible wedges by raising a chassis alarm on the Routing Engine.
- Verifies that the following datapath error detections are enabled for the IOC2, IOC3, and SPC2 cards:
  - Wedge detection for XM driver
  - Wedge detection for LU driver
  - Wedge detection for XL driver
  - Wedge detection for TOE driver (SPC2 only)
  - Wedge detection for host loopback (SPC2 only)

All datapath wedge conditions are detected and reported within 5 seconds. Each error detecting module records and reports the state and history of its identifiable wedge conditions.

Release History Table

Release	Description
15.1X49-D50	Starting with Junos OS Release 15.1-X49-D50 and Junos OS Release 17.3R1, the error management enhancements are supported on IOC2 and IOC3 I/O cards (IOCs) and SPC2 Services Processing Cards (SPCs).
15.1X49-D30	Starting with Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, error detection enhancements are provided that detect additional errors on IOCs and SPCs and provide enhanced error management.

Release History Table

Release	Description
15.1X49-D50	Starting with Junos OS Release 15.1-X49-D50 and Junos OS Release 17.3R1, the error management enhancements are supported on IOC2 and IOC3 I/O cards (IOCs) and SPC2 Services Processing Cards (SPCs).
15.1X49-D30	Starting with Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, error detection enhancements are provided that detect additional errors on IOCs and SPCs and provide enhanced error management.
12.1X47-D20	Starting with Junos OS Release 12.1X47-D20 and Junos OS Release 17.3R1, the system log includes information about NAT rule type.

RELATED DOCUMENTATION

Monitoring X2 Traffic | 200

Monitoring X2 Traffic

IN THIS SECTION

- Understanding X2 Traffic Monitoring | 201
- Example: Configuring a Mirror Filter for X2 Traffic Monitoring | 204

This topic covers X2 traffic monitoring on SRX Series devices.

Understanding X2 Traffic Monitoring

IN THIS SECTION

- X2 Traffic Monitoring Overview | 201
- Limitations of X2 Traffic Monitoring | 203
- X2 Traffic Terminology | 203

In an LTE mobile network, SRX Series devices act as secure gateways connecting Evolved Node Bs (eNodeBs) for signal handover, monitoring, and radio coverage. SRX Series devices use IPsec tunnels to connect eNodeBs. The user plane and control plane traffic that flows from one eNodeB to the other eNodeB is called the X2 traffic.

X2 Traffic Monitoring Overview

The X2 traffic passing through IPsec tunnels is encrypted. Because of this, mobile network operators need a way to monitor X2 traffic so that they can debug handover issues across eNodeBs. The Junos OS implementation allows monitoring of the X2 traffic by snooping into the cleartext X2 traffic as it flows through the SRX Series device coming out of one IPsec tunnel and going into the other IPsec tunnel—after traffic is decrypted and before it is encrypted again.

Figure 20 on page 202 shows the flow of X2 traffic within the SRX Series device. As the traffic reaches the SRX Series device on one st0.x interface, it gets decrypted. Then it is encrypted and forwarded to the destination eNodeB through its dedicated st0.y interface. Snooping is performed on the decrypted X2 traffic on the SRX Series device.

Figure 20: SRX Series Device in an LTE Mobile Network

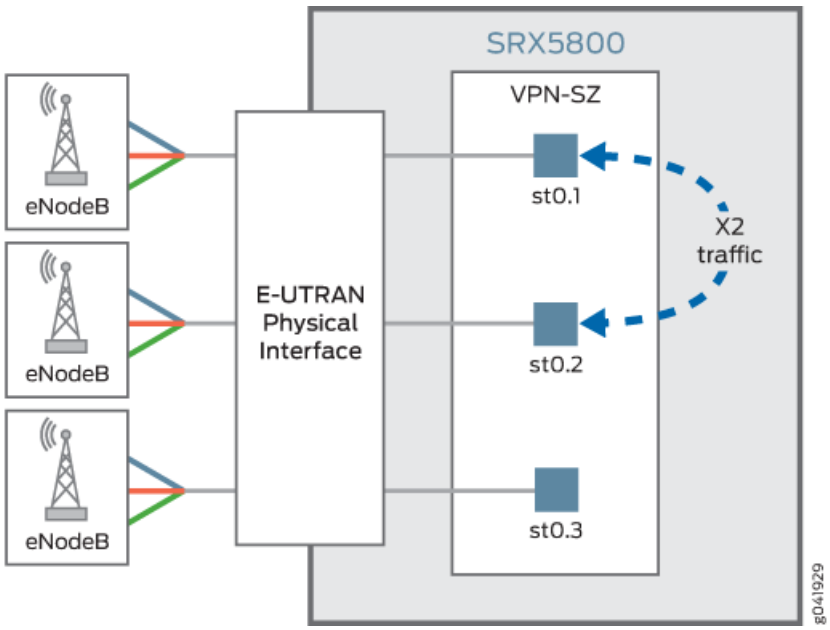
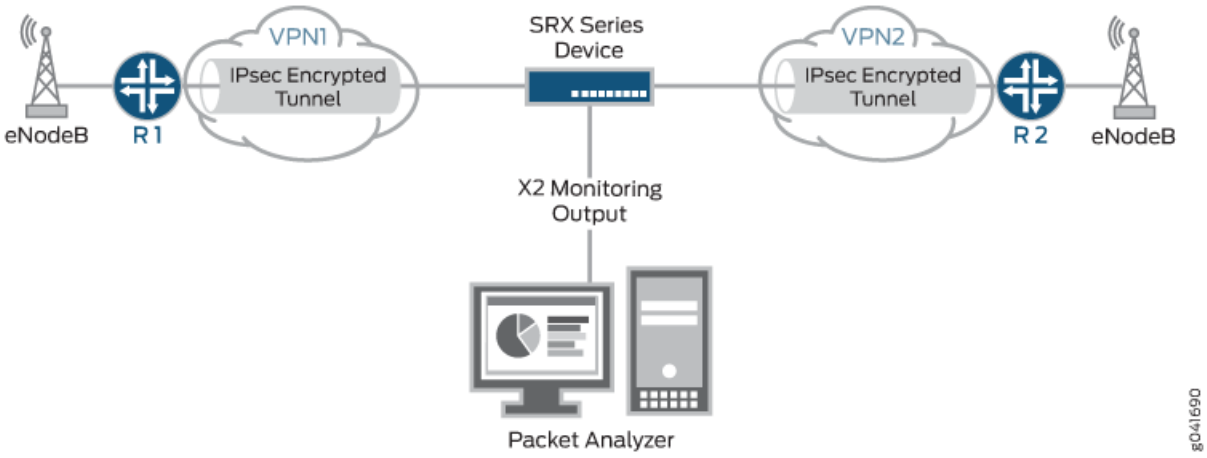


Figure 21 on page 202 shows a mobile operators network with an SRX Series device providing IPsec tunnel connection between the two eNodeBs. The SRX Series device is connected to a packet analyzer (also called a *sniffing* device) that is used for collecting and monitoring the X2 traffic. The IPsec tunnel from each eNodeB terminates on a dedicated secure tunnel interface on the SRX Series device. Inbound traffic coming out of the IPsec tunnel is decrypted while outbound traffic leaving the device is encrypted.

Figure 21: Monitoring X2 Traffic



To monitor the X2 traffic, you can configure up to 15 different mirror filters that specify unique sets of parameters against which traffic is matched. The filtered packets are duplicated and sent to a physical interface. To allow the packet analyzer to capture the filtered packets, you specify the output interface on the SRX Series device and the MAC address of the packet analyzer. Because the output interface is

connected to the same Layer 2 network as the packet analyzer, once mirror filtering is turned on, the packet analyzer can collect and analyze the X2 traffic.

The SRX Series mirror filter feature is bidirectional, much like a session. X2 traffic flowing through an IPSec VPN that matches a mirror filter is mirrored and analyzed; traffic returning from those devices is also mirrored and analyzed.

Starting in Junos OS Release 18.4R1, if the output X2 interface of a mirror filter is configured for an st0 interface to filter traffic that you want to analyze, the packet is duplicated and encrypted by the IPsec tunnel bound to the st0 interface. This enhancement supports the SRX Series devices to send traffic mirrored from a port on an IPsec tunnel. Mirrored traffic includes unmodified Layer 3 headers.

> **NOTE:** Although there is no minimum required number of parameters for a mirror filter, please be mindful that if you specify too few criteria or accidentally commit an incomplete filter, an over-proportional amount of traffic flow through the system could be mirrored.

Limitations of X2 Traffic Monitoring

For X2 traffic in a chassis cluster setup, mirrored packets cannot traverse through the data link (fabric interface).

X2 Traffic Terminology

Table 15 on page 203 lists some X2 traffic related terms and their descriptions.

Table 15: X2 Traffic Terminology

Term	Description
Evolved packet core (EPC)	Main component of System Architecture Evolution (SAE) and is also known as the SAE core. The EPC supports the IP network and serves as the equivalent of a General Packet Radio Service (GPRS) network, using the mobility management entity (MME), Serving Gateway (SGW), and Packet Data Network Gateway (PGW) subcomponents.
Evolved Universal Terrestrial Radio Access Network (E-UTRAN)	A radio access network standard. E-UTRAN is a new air interface system. It provides higher data rates and lower latency and is optimized for packet data. It uses Orthogonal Frequency-Division Multiple Access (OFDMA) for the downlink and Single-carrier Frequency Division Multiple Access for the uplink.

Table 15: X2 Traffic Terminology *(continued)*

Term	Description
Evolved Node B (eNodeB)	A device connected to the mobile phone network that communicates directly with mobile handsets, like a base transceiver station in Global System for Mobile Communications (GSM) networks. An eNodeB is controlled by a radio network controller (RNC).
Long Term Evolution (LTE)	A standard for wireless communication of high-speed data for mobile phones and data terminals. It increases the capacity and speed using a different radio interface and makes core network improvements.
X2 interface	A point-to-point logical interface between two eNodeBs with the E-UTRAN. It supports the exchange of signaling information between two eNodeBs and supports the forwarding of protocol data units (PDUs) to the respective tunnel endpoints.
X2 Application Protocol (X2AP)	Protocol used by the X2 interface. It is used for handling the user equipment mobility within the E-UTRAN and provides the following functions: • Manages mobility and load • Reports general error situations • Sets and resets the X2 interface • Updates the eNodeB configuration

Example: Configuring a Mirror Filter for X2 Traffic Monitoring

IN THIS SECTION

- Requirements | **205**
- Overview | **205**
- Configuration | **206**
- Verification | **208**

This example shows how to configure a mirror filter to monitor X2 traffic between two eNodeBs in an LTE mobile network.

Requirements

Before you begin:

- Understand X2 traffic monitoring. .
- Configure the interfaces, security zones, security policies, and the route-based VPN tunnels to allow data to be securely transferred between the SRX Series device and the two eNodeBs.

Overview

As a network operator, you need a way to monitor the X2 traffic to debug any handover issues across eNodeBs. The mirror filter feature allows you to do that. Traffic coming out of an IPsec tunnel is decrypted, mirrored and analyzed, and then encrypted again to go into the outbound IPsec tunnel.

More specifically, traffic that matches a mirror filter is mirrored and sent to an output interface that is connected to a packet analyzer (also called a *sniffing* device). The packet analyzer analyzes the X2 traffic, allowing you to monitor it. Then the traffic is encrypted again before it is sent to the outbound IPsec tunnel.

The SRX Series mirror filter feature is bidirectional, much like a session. X2 traffic flowing through an IPSec VPN that matches a mirror filter is mirrored and analyzed; traffic returning from those devices is also mirrored and analyzed.

To use the mirror filter feature to monitor X2 traffic, you configure mirror filters. You can configure up to 15 different mirror filters to be used concurrently to filter for various kinds of traffic. Each mirror filter contains a set of parameters and their values against which traffic is matched.

> **NOTE:** Although there is no minimum required number of parameters for a mirror filter, please be mindful that if you specify too few criteria or accidentally commit an incomplete filter, an over-proportional amount of traffic flow through the system could be mirrored.

A mirror filter can contain some or all of the following parameters to filter traffic:

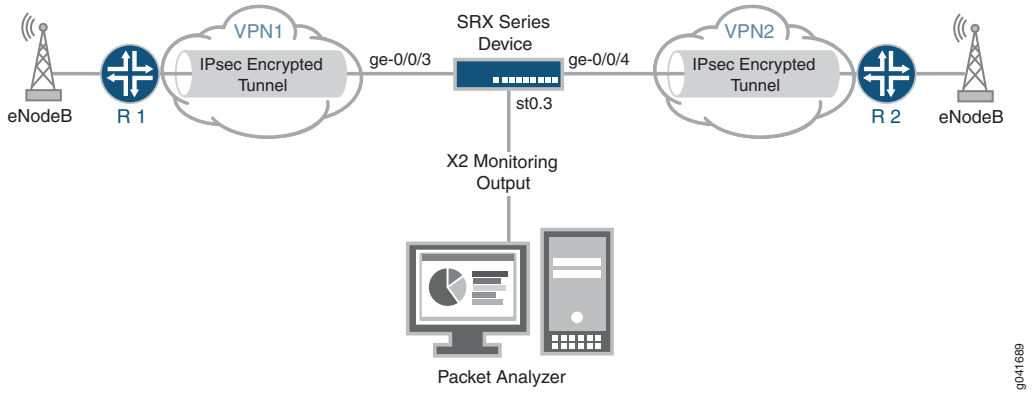
- destination IP address prefix
- destination port
- IP protocol
- source IP address prefix
- source port
- incoming and outgoing interfaces

You also specify the output interface and the MAC address of the packet analyzer as part of the configuration.

In this example, an SRX Series device uses IPsec tunnels to connect two eNodeBs in an LTE mobile network. The example configures a mirror filter called traffic-https.

Figure 22 on page 206 shows the SRX Series device connecting to the eNodeBs using IPsec tunnels. The SRX Series device is also connected to a packet analyzer.

Figure 22: Configuring Mirror Filters for X2 Traffic Monitoring



In this example, all HTTPS traffic is analyzed whose destination is to devices with IP addresses that have the prefix 203.0.113.0/24 and for which the destination port 443 is used, the default port for HTTPS traffic. Packets that match the traffic-https filter are *mirrored* and sent through the output interface ge-0/0/5 to the packet analyzer with the MAC address 00:50:56:87:20:5E. Returning traffic from these devices is also monitored.

> **NOTE:** The output interface for mirror filter is that of the packet analyzer, which is why the HTTP protocol is used.
>
> The output interface for the packet analyzer uses the HTTP protocol.

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security forwarding-options mirror-filter traffic-https
set security forwarding-options mirror-filter traffic-https destination-port 443
```

```
set security forwarding-options mirror-filter traffic-https destination-prefix 203.0.113.0/24
set security forwarding-options mirror-filter traffic-https protocol 6
set security forwarding-options mirror-filter traffic-http output interface ge-0/0/5
set security forwarding-options mirror-filter traffic-http output destination-mac 00:50:56:87:20:5E
```

Step-by-Step Procedure

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

To configure a mirror filter for monitoring X2 traffic:

1. Create a mirror filter called traffic-https.

   ```
   [edit]
   user@host# edit security forwarding-options mirror-filter traffic-https
   ```

2. Specify the mirror filter parameters against which traffic is matched.

   ```
   [edit security forwarding-options mirror-filter traffic-https]
   user@host# set destination-port 443
   user@host# set destination-prefix 203.0.113.0/24
   user@host# set protocol 6
   ```

3. Specify the output interface for the mirrored packets to be sent to the packet analyzer.

   ```
   [edit security forwarding-options mirror-filter traffic-https]
   user@host# set output interface ge-0/0/5
   ```

4. Specify the MAC address of the packet analyzer as a destination for all mirrored packets, that is, those packets that match the mirror filters.

   ```
   [edit security forwarding-options mirror-filter traffic-https]
   user@host# set output destination-mac 00:50:56:87:20:5E
   ```

Results

From configuration mode, confirm your configuration by entering the **show security forwarding-options** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
user@host# show security forwarding-options
  mirror-filter traffic-https {
    protocol 6;
    destination-port 443;
    destination-prefix 203.0.113.0/24;
    output {
      interface ge-0/0/5;
      destination-mac 00:50:56:87:20:5E;
    }
  }
```

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

Verification

Confirm that the configuration is working properly.

Verifying the Status of Mirror Filter

Purpose

Verify that mirror filter is active or not.

Action

From operational mode, enter the **show security forward-options mirror-filter** command for the specific mirror filter.

`user@host>` **show security forward-options mirror-filter traffic-https**

```
Security mirror status

    mirror-filter-name: traffic-https
    protocol: 6
   destination-port: 443
    destination-prefix 203.0.113.0/24
    filter-counters: 2
    output-counters: 2
```

Meaning

The output provides the mirror filter status. It shows that a mirror filter called traffic-https is active. The traffic-https mirror filter specifies the protocol, destination prefix, and destination port that traffic must match in order for it to be mirrored and analyzed.

This output shows that two packets were mirrored.

5
CHAPTER

Packet Based Forwarding

Packet-Based Forwarding | **211**

Reverse Route Packet Mode using Virtual Router | **246**

Express Path | **249**

Packet-Based Forwarding

IN THIS SECTION

- Understanding Packet-Based Processing | 211
- Understanding Selective Stateless Packet-Based Services | 212
- Selective Stateless Packet-Based Services Configuration Overview | 214
- Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding | 216
- Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding | 228
- Understanding Session Cache | 241
- Understanding Symmetric Fat IPsec Tunnel | 245

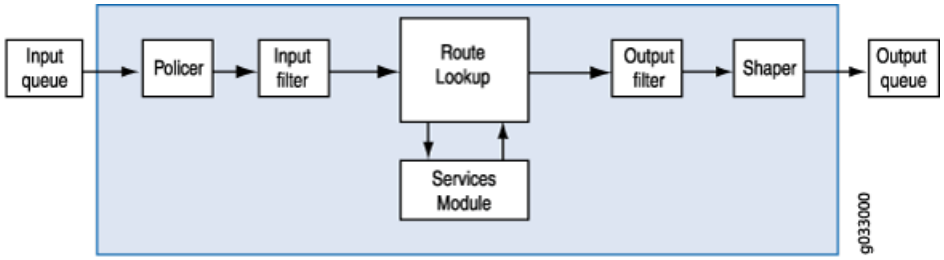
An SRX device operate in two different modes: packet mode and flow mode. In flow mode, SRX processes all traffic by analyzing the state or session of traffic. This is also called stateful processing of traffic. In packet mode, SRX processes the traffic on a per-packet basis. This is also known as stateless processing of traffic.

Understanding Packet-Based Processing

Packets that enter and exit a Juniper Networks device running Junos OS can undergo packet-based processing. Packet-based, or stateless, packet processing treats packets discretely. Each packet is assessed individually for treatment. Stateless packet-based forwarding is performed on a packet-by-packet basis without regard to flow or state information. Each packet is assessed individually for treatment.

Figure 23 on page 212 shows the traffic flow for packet-based forwarding.

Figure 23: Traffic Flow for Packet-Based Forwarding



As packets enter the device, classifiers, filters and policers are applied to it. Next, the egress interface for the packet is determined through a route lookup. Once the egress interface for the packet is found, filters are applied and the packet is sent to the egress interface where it is queued and scheduled for transmission.

Packet-based forwarding does not require any information about either previous or subsequent packets that belong to a given connection, and any decision to allow or deny traffic is packet specific. This architecture has the benefit of massive scaling because it forwards packets without keeping track of individual flows or state.

Starting with Junos OS Release 15.1X49-D100, for the SRX100, SRX110, SRX210, SRX220, SRX240, SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX650, the maximum capture size for packet captures is expanded to 1520 bytes to allow for captures of 1500 bytes of data and the 12-byte Juniper Ethernet header."

Understanding Selective Stateless Packet-Based Services

Selective stateless packet-based services allow you to use both flow-based and packet-based forwarding simultaneously on a system. You can selectively direct traffic that requires packet-based, stateless forwarding to avoid stateful flow-based forwarding by using stateless firewall filters, also known as access control lists (ACLs). The traffic not so directed follows the default flow-based forwarding path. Bypassing flow-based forwarding can be useful for traffic for which you explicitly want to avoid flow session-scaling constraints.

By default, Juniper Networks Security devices running Junos OS use flow-based forwarding. Selective stateless packet-based services allows you to configure the device to provide only packet-based processing for selected traffic based on input filter terms. Other traffic is processed for flow-based forwarding. Bypassing flow-based forwarding is useful for deployments where you want to avoid session-scaling constraints and session creation and maintenance costs.

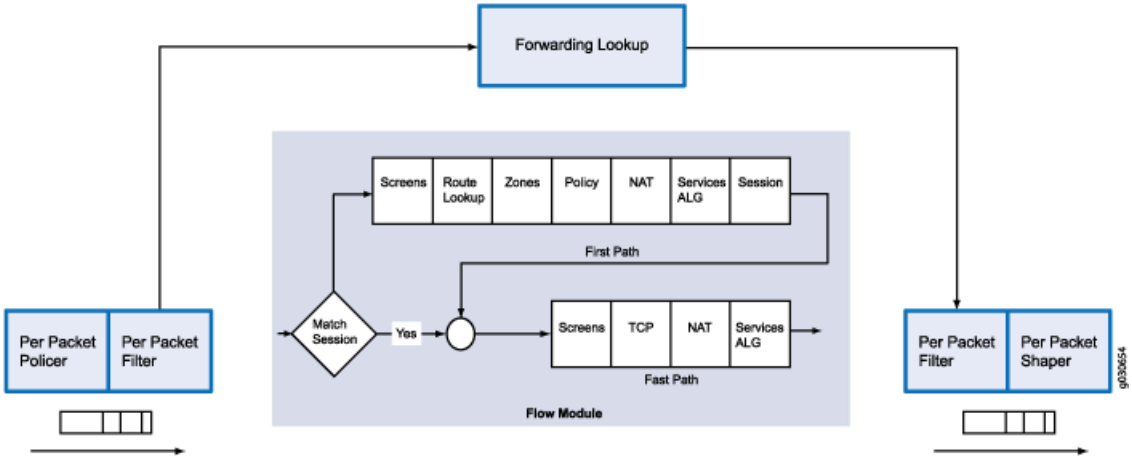
When you configure the device for selective stateless packet-based processing, packets entering the system are treated differently depending on certain conditions:

- If a packet satisfies matching conditions specified in input filter terms, it is marked for packet mode and all configured packet mode features are applied to it. No flow-based security features are applied. It bypasses them.

- If a packet has not been flagged for packet-mode, it undergoes normal processing. All services except for MPLS can be applied to this traffic.

Figure 24 on page 213 shows traffic flow with selective stateless packet-based services bypassing flow-based processing.

Figure 24: Traffic Flow with Selective Stateless Packet-Based Services



When the packet comes in on an interface, the input packet filters configured on the interface are applied.

- If the packet matches the conditions specified in the firewall filter, a **packet-mode** action modifier is set to the packet. The packet-mode action modifier updates a bit field in the packet key buffer—this bit field is used to determine if the flow-based forwarding needs to be bypassed. As a result, the packet with the packet-mode action modifier bypasses the flow-based forwarding completely. The egress interface for the packet is determined through a route lookup. Once the egress interface for the packet is found, filters are applied and the packet is sent to the egress interface where it is queued and scheduled for transmission.
- If the packet does not match the conditions specified in this filter term, it is evaluated against other terms configured in the filter. If, after all terms are evaluated, a packet matches no terms in a filter, the packet is silently discarded. To prevent packets from being discarded, you configure a term in the filter specifying an action to accept all packets.

A defined set of stateless services is available with selective stateless packet-based services:

- IPv4 routing (unicast and multicast protocols)
- Class of service (CoS)
- Link fragmentation and interleaving (LFI)
- Generic routing encapsulation (GRE)
- Layer 2 switching
- Multiprotocol Label Switching (MPLS)

- Stateless firewall filters
- Compressed Real-Time Transport Protocol (CRTP)

Although traffic requiring MPLS services must be processed in packet mode, under some circumstances it might be necessary to concurrently apply certain services to this traffic that can only be provided in flow mode, such as stateful inspection, NAT, and IPsec. To direct the system to process traffic in both flow and packet modes, you must configure multiple routing instances connected through a tunnel interface. One routing instance must be configured to process the packets in flow mode and the other routing instance must be configured to process the packets in packet mode. When you use a tunnel interface to connect routing instances, traffic between those routing instances is injected again into the forwarding path and it can then be reprocessed using a different forwarding method.

Selective Stateless Packet-Based Services Configuration Overview

This feature is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices. You configure selective stateless packet-based services using the stateless firewall filters, also known as access control lists (ACLs). You classify traffic for packet-based forwarding by specifying match conditions in the firewall filters and configure a **packet-mode** action modifier to specify the action. Once match conditions and actions are defined, firewall filters are applied to relevant interfaces.

To configure a firewall filter:

1. Define the address family—First define the address family of the packets that a firewall filter matches. To define the family name, specify **inet** to filter IPv4 packets. Specify **mpls** to filter MPLS packets. Specify **ccc** to filter Layer 2 switching cross-connects.

2. Define terms—Define one or more terms that specify the filtering criteria and the action to take if a match occurs. Each term consists of two components—match conditions and actions.

   - Match conditions—Specify certain characteristics that the packet must match for the action to be performed. You can define various match conditions, such as the IP source address field, IP destination address field, and IP protocol field.
   - Action—Specify what is to be done with the packet if it matches the match conditions. Possible actions are to accept, discard, or reject a packet; go to the next term; or take no action.

     You can specify only one **action** (or omit it) in a term, but you can specify any combination of action modifiers with it. Action modifiers include a default **accept** action. For example, if you specify an action modifier and do not specify an action, the specified action modifier is implemented and the packet is accepted.

     The **packet-mode** action modifier specifies traffic to bypass flow-based forwarding. Like other action modifiers, you can configure the **packet-mode** action modifier along with other actions, such as **accept** or **count**.

3. Apply firewall filters to interfaces—Apply the firewall filter to the interface to have the firewall filter take effect.

When the packet comes in on an interface, the input packet filters configured on the interface are applied. If the packet matches the specified conditions and **packet-mode** action is configured, the packet bypasses the flow-based forwarding completely.

When configuring filters, be mindful of the order of the terms within the firewall filter. Packets are tested against each term in the order in which it is listed in the configuration. When the first matching conditions are found, the action associated with that term is applied to the packet and the evaluation of the firewall filter ends, unless the **next term** action modifier is included. If the **next term** action is included, the matching packet is then evaluated against the next term in the firewall filter; otherwise, the matching packet is not evaluated against subsequent terms in the firewall filter.

When configuring firewall filters for selective stateless packet-based services:

- Accurately identify traffic that needs to bypass flow to avoid unnecessary packet drops.
- Make sure to apply the firewall filter with packet-mode action on all interfaces involved in the packet-based flow path.
- Make sure to configure host-bound TCP traffic to use flow-based forwarding—exclude this traffic when specifying match conditions for the firewall filter term containing the **packet-mode** action modifier. Any host-bound TCP traffic configured to bypass flow is dropped. Asynchronous flow-mode processing is not supported with selective stateless packet-based services.
- Configure input packet filters (not output) with the **packet-mode** action modifier.

> **NOTE:** Nested firewall filters (configuring a filter within the term of another filter) are not supported with selective stateless packet-based services.

Some typical deployment scenarios where you can configure selective stateless packet-based services are as follows:

- Traffic flow between private LAN and WAN interfaces, such as for Intranet traffic, where end-to-end forwarding is packet-based
- Traffic flow between private LAN and not-so-secure WAN interfaces, where traffic uses packet-based and flow-based forwarding for secure and not so secure traffic respectively
- Traffic flow between the private LAN and WAN interface with failover to flow-based IPsec WAN when the private WAN link is down
- Traffic flow from flow-based LAN to packet-based MPLS WAN

Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding

IN THIS SECTION

- Requirements | 216
- Overview | 216
- Configuration | 217
- Verification | 224

This example shows how to configure selective stateless packet-based services for end-to-end packet-based forwarding. This feature is supported on the SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices

Requirements

Before you begin:

- Understand how to configure stateless firewall filters.
- Establish basic connectivity. .

Overview

In this example, you configure the IP addresses for the interfaces on each of the devices. For R0 it is 10.1.1.2/24 ; for R1 they are 10.1.1.1/24, 10.2.1.1/24, and 203.0.113.1/30; for R2 it is 203.0.113.2/30; and for R3 it is 10.2.1.2/24. You create static routes and associate next-hop addresses for the devices as follows: R0 is 10.1.1.2, R1 is 198.51.100.2, R2 is 203.0.113.1, and R3 is 10.2.1.1.

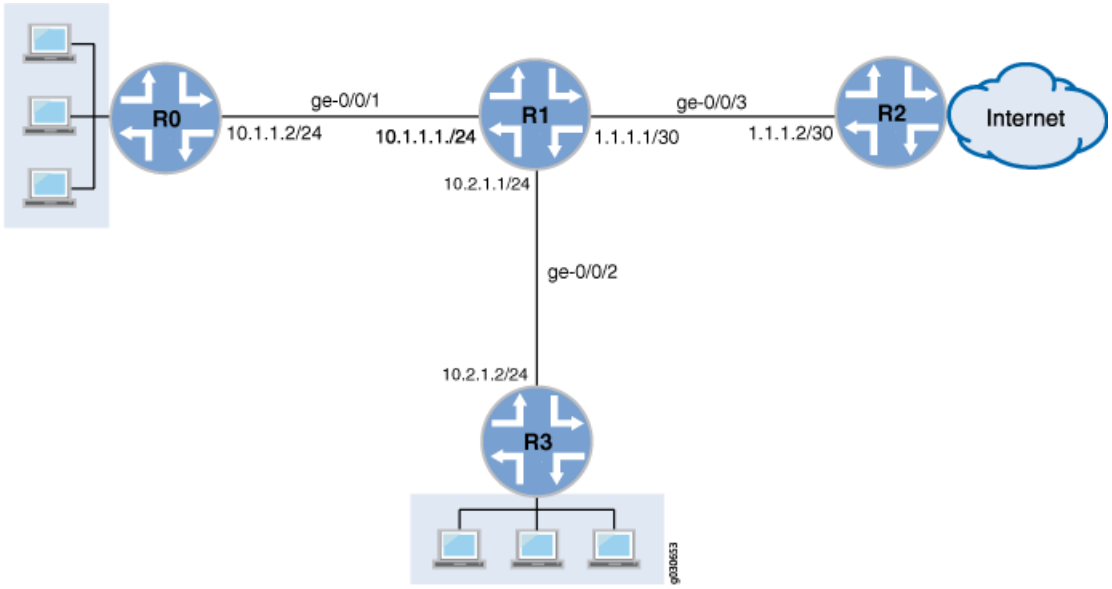
Then on device R1 you configure a zone called untrust and assign it to interface ge-0/0/3. You also create a zone called trust and assign interfaces ge-0/0/1 and ge-0/0/2 to it. You configure trust and untrust zones to allow all supported application services as inbound services. You allow traffic from any source address, destination address, and application to pass between the zones.

You then create the firewall filter bypass-flow-filter and define the terms bypass-flow-term-1 and bypass-flow-term-2 that match the traffic between internal interfaces ge-0/0/1 and ge-0/0/2 and that contain the packet-mode action modifier. You define the term accept-rest to accept all remaining traffic. Finally, you apply the firewall filter bypass-flow-filter to internal interfaces ge-0/0/1 and ge-0/0/2 (not

on the external interface). As a result, all internal traffic bypasses flow-based forwarding and the traffic to and from the Internet does not bypass flow-based forwarding.

Figure 25 on page 217 shows the network topology used in this example.

Figure 25: Intranet Traffic Using End-to-End Packet-Based Services



Your company's branch offices are connected to each other through a private WAN. For this internal traffic, packet forwarding is required because security is not an issue. Hence for this traffic, you decide to configure selective stateless packet-based services to bypass flow-based forwarding. The remaining traffic, to and from the Internet, uses flow-based forwarding.

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, and then enter **commit** from configuration mode.

```
{device R0}
[edit]
set interfaces ge-0/0/1 description "Internal 1" unit 0 family inet address 10.1.1.2/24
set routing-options static route 0.0.0.0/0 next-hop 10.1.1.1
```

```
{device R1}
set interfaces ge-0/0/1 description "Internal 1" unit 0 family inet address 10.1.1.1/24
set interfaces ge-0/0/2 description "Internal 2" unit 0 family inet address 10.2.1.1/24
```

```
set interfaces ge-0/0/3 description "Internet" unit 0 family inet address 203.0.113.1/30
set routing-options static route 0.0.0.0/0 next-hop 203.0.113.2
set security zones security-zone untrust interfaces ge-0/0/3
set security zones security-zone trust interfaces ge-0/0/1
set security zones security-zone trust interfaces ge-0/0/2
set security zones security-zone trust host-inbound-traffic system-services all
set security zones security-zone untrust host-inbound-traffic system-services all
set security policies from-zone trust to-zone untrust policy Internet-traffic match source-address any
   destination-address any application any
set security policies from-zone trust to-zone untrust policy Internet-traffic then permit
set security policies from-zone untrust to-zone trust policy Incoming-traffic match source-address any
   destination-address any application any
set security policies from-zone untrust to-zone trust policy Incoming-traffic then permit
set security policies from-zone trust to-zone trust policy Intrazone-traffic match source-address any
   destination-address any application any
set security policies from-zone trust to-zone trust policy Intrazone-traffic then permit
set firewall family inet filter bypass-flow-filter term bypass-flow-term-1 from source-address 10.1.1.0/24
set firewall family inet filter bypass-flow-filter term bypass-flow-term–1 from destination-address 10.2.1.0/24
set firewall family inet filter bypass-flow-filter term bypass-flow-term-1 then packet-mode
set firewall family inet filter bypass-flow-filter term bypass-flow-term-2 from source-address 10.2.1.0/24
set firewall family inet filter bypass-flow-filter term bypass-flow-term-2 from destination-address 10.1.1.0/24
set firewall family inet filter bypass-flow-filter term bypass-flow-term-2 then packet-mode
set firewall family inet filter bypass-flow-filter term accept-rest then accept
set interfaces ge-0/0/1 description "Internal 1" unit 0 family inet filter input bypass-flow-filer
set interfaces ge-0/0/2 description "Internal 2" unit 0 family inet filter input bypass-flow-filer
```

```
{device R2}
set interfaces ge-0/0/3 description "Internet" unit 0 family inet address 10.1.1.2/30
set routing-options static route 0.0.0.0/0 next-hop 10.1.1.1
```

```
{device R3}
[edit]
set interfaces ge-0/0/2 description "Internal 2" unit 0 family inet address 10.21.1.2/24
set routing-options static route 0.0.0.0/0 next-hop 10.2.1.1
```

Step-by-Step Procedure

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

To configure selective stateless packet-based services for end-to-end packet-based forwarding:

1. Configure the IP addresses for the interfaces on devices R0, R1, R2, and R3.

```
{device R0}
[edit]
user@host# set interfaces ge-0/0/1 description "Internal 1" unit 0 family inet address 10.1.1.2/24
```

```
{device R1}
[edit]
user@host# set interfaces ge-0/0/1 description "Internal 1" unit 0 family inet address 10.1.1.1/24
user@host# set interfaces ge-0/0/2 description "Internal 2" unit 0 family inet address 10.2.1.1/24
user@host# set interfaces ge-0/0/3 description "Internet" unit 0 family inet address 203.0.113.1/30
```

```
{device R2}
[edit]
user@host# set interfaces ge-0/0/3 description "Internet" unit 0 family inet address 203.0.113.1/30
```

```
{device R3}
[edit]
user@host# set interfaces ge-0/0/2 description "Internal 2" unit 0 family inet address 10.2.1.2/24
```

2. Create static routes and associate the appropriate next-hop addresses for devices R0, R1, R2, and R3.

```
{device R0}
[edit]
user@host# set routing-options static route 0.0.0.0/0 next-hop 10.1.1.1
```

```
{device R1}
[edit]
user@host# set routing-options static route 0.0.0.0/0 next-hop 203.0.113.1
```

```
{device R2}
[edit]
user@host# set routing-options static route 0.0.0.0/0 next-hop 203.0.113.2
```

```
{device R3}
[edit]
user@host# set routing-options static route 0.0.0.0/0 next-hop 10.2.1.1
```

3. Configure security zones and assign interfaces.

```
{device R1}
[edit]
user@host# set security zones security-zone untrust interfaces ge-0/0/3
user@host# set security zones security-zone trust interfaces ge-0/0/1
user@host# set security zones security-zone trust interfaces ge-0/0/2
```

4. Configure application services for zones.

```
{device R1}
[edit]
user@host# set security zones security-zone trust host-inbound-traffic system-services all
user@host# set security zones security-zone untrust host-inbound-traffic system-services all
```

5. Configure a security policy

```
{device R1}
[edit]
user@host# set security policies from-zone trust to-zone untrust policy Internet-traffic match source-address
   any destination-address any application any
user@host# set security policies from-zone trust to-zone untrust policy Internet-traffic then permit
user@host# set security policies from-zone untrust to-zone trust policy Incoming-traffic match source-address
   any destination-address any application any
user@host# set security policies from-zone untrust to-zone trust policy Incoming-traffic then permit
user@host# set security policies from-zone trust to-zone trust policy Intrazone-traffic match source-address
   any destination-address any application any
user@host# set security policies from-zone trust to-zone trust policy Intrazone-traffic then permit
```

6. Create a firewall filter and define terms for all the packet-based forwarding traffic.

```
{device R1}
[edit]
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term-1 from source-address
   10.1.1.0/24
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term–1 from destination-address
   10.2.1.0/24
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term-1 then packet-mode
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term-2 from source-address
   10.2.1.0/24
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term-2 from destination-address
   10.1.1.0/24
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term-2 then packet-mode
```

7. Specify another term for the remaining traffic.

```
{device R1}
[edit]
user@host# set firewall family inet filter bypass-flow-filter term accept-rest then accept
```

8. Apply the firewall filter to relevant interfaces.

```
{device R1}
[edit]
user@host# set interfaces ge-0/0/1 description "Internal 1" unit 0 family inet filter input bypass-flow-filer
user@host# set interfaces ge-0/0/2 description "Internal 2" unit 0 family inet filter input bypass-flow-filer
```

Results

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

```
{device R0}
[edit]
user@host# show interfaces
ge-0/0/1 {
   description “Internal 1”
   unit 0 {
      family inet {
         address 10.1.1.2/24
      }
   }
}
```

```
{device R0}
[edit]
user@host# show routing-options
static {
   route 0.0.0.0/0 next-hop 10.1.1.1;
}
```

```
{device R2}
[edit]
user@host# show interfaces
```

```
ge-0/0/3 {
    description “Internet”
    unit 0 {
        family inet {
            address 203.0.113.2/30;
        }
    }
}
```

```
{device R2}
[edit]
user@host# show routing-options
static {
    route 0.0.0.0/0 next-hop 203.0.113.1;
}
```

```
{device R3}
[edit]
user@host# show interfaces
ge-0/0/2 {
    description “Internal 2”
    unit 0 {
        family inet {
            address 10.2.1.2/24;
        }
    }
}
```

```
{device R3}
user@host# show routing-options
static {
    route 0.0.0.0/0 next-hop 10.2.1.1;
}
```

```
{device R1}
[edit]
user@host# show interfaces
ge-0/0/1 {
    description “internal 1”
    unit 0 {
        family inet {
```

```
            filter {
                input bypass-flow-filter;
                }
            address 10.1.1.1/24;
        }
    }
}
ge-0/0/2 {
    description “Internal 2”
    unit 0 {
        family inet {
            filter {
                input bypass-flow-filter;
                }
            address 10.2.1.1/24;
        }
    }
}
ge-0/0/3 {
    description “Internet”
    unit 0 {
        family inet {
            address 203.0.113.1/30;
        }
    }
}
{device R1}
[edit]
user@host# show routing-options
static {
    route 0.0.0.0/0 next-hop 203.0.113.1;
}
{device R1}
[edit]
user@host# show firewall
family inet {
    filter bypass-flow-filter {
        term bypass-flow-term-1 {
            from {
                source-address {
                    10.1.1.0/24;
                }
                destination-address {
                    10.2.1.0/24;
```

```
            }
          }
          then packet-mode;
        }
        term bypass-flow-term-2 {
          from {
            source-address {
              10.2.1.0/24;
            }
            destination-address {
              10.1.1.0/24;
            }
          }
          then packet-mode;
        }
        term accept-rest {
          then accept;
        }
      }
    }
```

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

Verification

IN THIS SECTION

- Verifying the End-to-End Packet-Based Configuration | **224**
- Verifying Session Establishment on Intranet Traffic | **225**
- Verifying Session Establishment on Internet Traffic | **227**

Confirm that the configuration is working properly.

Verifying the End-to-End Packet-Based Configuration

Purpose

Verify that the selective stateless packet-based services are configured.

Action

From configuration mode, enter the **show interfaces**, **show routing-options**, **show security zones**, **show security policies**, and **show firewall** commands.

Verify that the output shows the intended configuration of the firewall filter, interfaces, and policies.

Verify that the terms are listed in the order in which you want the packets to be tested. You can move terms within a firewall filter by using the **insert** command.

Verifying Session Establishment on Intranet Traffic

Purpose

Verify that sessions are established when traffic is transmitted to interfaces within the Intranet.

Action

To verify that sessions are established, perform the following tasks:

1. On device **R1**, enter the operational mode **clear security flow session all** command to clear all existing security flow sessions.

2. On device **R0**, enter the operational mode **ping** command to transmit traffic to device **R3**.

3. On device **R1**, with traffic transmitting from devices **R0** to **R3** through **R1**, enter the operational mode **show security flow session** command.

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

NOTE: To verify established sessions, make sure to enter the **show security flow session** command while the **ping** command is sending and receiving packets.

Starting in Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, the session flow summaries include CP session IDs.

{device R0}

user@host> **ping 192.0.2.2 -c 10**

```
PING 192.0.2.2 (192.0.2.2) 56(84) bytes of data.
64 bytes from 192.0.2.2: icmp_seq=1 ttl=63 time=6.07 ms
64 bytes from 192.0.2.2: icmp_seq=2 ttl=63 time=4.24 ms
64 bytes from 192.0.2.2: icmp_seq=3 ttl=63 time=2.85 ms
64 bytes from 192.0.2.2: icmp_seq=4 ttl=63 time=6.14 ms
...
```

{device R1}
user@host>**show security flow session**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000077, Policy name: Internet-traffic/5, Timeout: 2, Valid
  In: 198.51.100.1/3 --> 192.0.2.2/32055;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 410000198
  Out: 192.0.2.2/32055 --> 198.51.100.1/3;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes:
84, CP Session ID: 410000198
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Session ID: 420000079, Policy name: Internet-traffic/5, Timeout: 2, Valid
  In: 198.51.100.1/5 --> 192.0.2.2/32055;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000163
  Out: 192.0.2.2/32055 --> 198.51.100.1/5;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes:
84, CP Session ID: 420000163
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000090, Policy name: Internet-traffic/5, Timeout: 4, Valid
  In:198.51.100.1/7 --> 192.0.2.2/32055;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000088
  Out: 192.0.2.2/32055 --> 198.51.100.1/7;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes:
84, CP Session ID: 430000088
Total sessions: 1
```

The output shows traffic transmitting from **R0** to **R3** and no sessions are established. In this example, you applied the **bypass-flow-filter** with the **packet-mode** action modifier on interfaces **Internal 1** and **Internal 2** for your company's Intranet traffic. This output verifies that the traffic between the two interfaces is correctly bypassing flow-based forwarding and hence no sessions are established.

Verifying Session Establishment on Internet Traffic

Purpose

Verify that sessions are established when traffic is transmitted to the Internet.

Action

To verify that traffic to the Internet is using flow-based forwarding and sessions are established, perform the following tasks:

1. On device **R1**, enter the operational mode **clear security flow session all** command to clear all existing security flow sessions.

2. On device **R0**, enter the operational mode **ping** command to transmit traffic to device **R2**.

3. On device **R1**, with traffic transmitting from **R0** to **R2** through **R1**, enter the operational mode **show security flow session** command.

> **NOTE:** To verify established sessions, make sure to enter the **show security flow session** command while the **ping** command is sending and receiving packets.

{device R0}
user@host> **ping 203.0.113.6**

```
PING 203.0.113.6 (203.0.113.6): 56 data bytes
64 bytes from 203.0.113.6: icmp_seq=0 ttl=63 time=2.326 ms
64 bytes from 203.0.113.6: icmp_seq=1 ttl=63 time=2.569 ms
64 bytes from 203.0.113.6: icmp_seq=2 ttl=63 time=2.565 ms
64 bytes from 203.0.113.6: icmp_seq=3 ttl=63 time=2.563 ms
64 bytes from 203.0.113.6: icmp_seq=4 ttl=63 time=2.306 ms
64 bytes from 203.0.113.6: icmp_seq=5 ttl=63 time=2.560 ms
64 bytes from 203.0.113.6: icmp_seq=6 ttl=63 time=4.130 ms
64 bytes from 203.0.113.6: icmp_seq=7 ttl=63 time=2.316 ms
...
```

```
{device R1}
user@host> show security flow session
```

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

The output shows traffic transmitting from devices **R0** to **R1** and established sessions. In this example, you did not apply the **bypass-flow-filter** with the **packet-mode** action modifier on interface **Internet** for your company's Internet traffic. This output verifies that the traffic to the Internet is correctly using flow-based forwarding and hence sessions are established.

Transmit traffic from device **R3** to **R2** and use the commands in this section to verify established sessions.

Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding

IN THIS SECTION

- Requirements | **229**
- Overview | **229**
- Configuration | **230**
- Verification | **237**

This example shows how to configure selective stateless packet-based services for packet-based to flow-based forwarding. This feature is supported on SRX300, SRX320, SRX340, SRX345, SRX550M, and SRX1500 devices.

Requirements

Before you begin:

- Understand how to configure stateless firewall filters.
- Establish basic connectivity. .

Overview

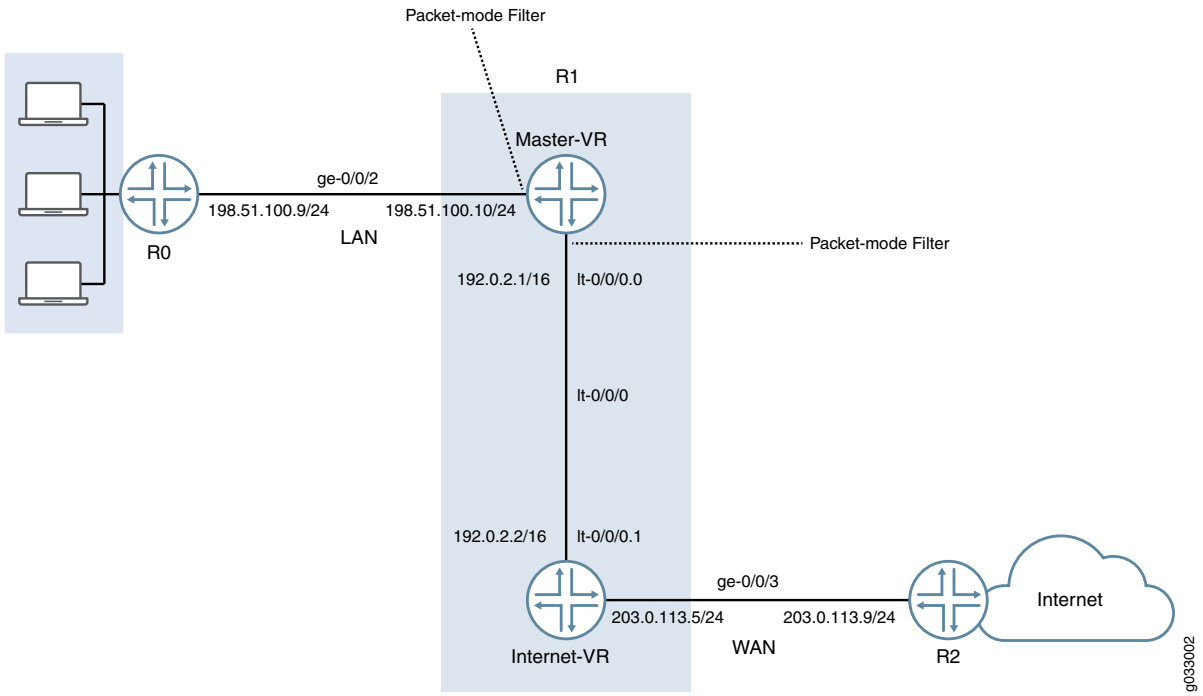
In this example, you configure the IP addresses for the interfaces on each of the devices. For device R0 as 198.51.100.9/24; for R1 the are198.51.100.10/24 and 203.0.113.5/24; and for R2 it is 203.0.113.9/24. On device R1, you set an internal service interface lt-0/0/0 between routing instances and configure a peer relationship between two virtual devices. You then create two security zones, Master-VR-zone and Internet-VR-zone, assign related interfaces to them, and configure them to allow all supported applications and protocols.

Then you configure policies and specify that all packets are permitted. You configure a virtual device routing instance Internet-VR and assign interfaces for flow-based forwarding. You enable OSPF on devices R0, R1, and R2. On Device R2, you configure the filter bypass-flow-filter with the term bypass-flow-term that contains the packet-mode action modifier. Because you have not specified any match conditions, this filter applies to all traffic that traverses the interfaces on which it is applied.

Finally, on device R1 you apply the firewall filter bypass-flow-filter to internal interfaces ge-0/0/2.0 and lt-0/0/0.0. You do not apply the filter to the interfaces associated with the Internet-VR routing instance. As a result, all traffic that traverses the LAN interfaces associated with the master routing instance uses packet-based forwarding and all traffic that traverses the Internet-VR routing instance uses flow-based forwarding.

Figure 26 on page 230 shows the network topology used in this example.

Figure 26: Selective Stateless Packet-Based Services for Packet-Based Forwarding



The interface facing the private LAN does not need any security services, but the interface facing the WAN needs security. In this example, you decide to configure both packet-based and flow-based forwarding for secure and not so secure traffic by configuring two routing instances—one handling the packet-based forwarding and the other handling the flow-based forwarding.

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, and then enter **commit** from configuration mode.

{device R0}

set interfaces description “Connect to Master VR” ge-0/0/2 unit 0 family inet address 198.51.100.9/24
set protocols ospf area 0.0.0.0 interface ge-0/0/2.0

{device R1}

set interfaces description “Connect to R0” ge-0/0/2 unit 0 family inet address 198.51.100.10/24
set interfaces description “Connect to R2” ge-0/0/3 unit 0 family inet address 203.0.113.5/24
set interfaces lt-0/0/0 unit 0 encapsulation frame-relay dlci 100 peer-unit 1 family inet address 192.0.2.1/16

```
set interfaces lt-0/0/0 unit 1 encapsulation frame-relay dlci 100 peer-unit 0 family inet address 192.0.2.2/16
set security zones security-zone Master-VR-zone host-inbound-traffic system-services all
set security zones security-zone Master-VR-zone host-inbound-traffic protocols all
set security zones security-zone Master-VR-zone interfaces ge-0/0/2.0
set security zones security-zone Master-VR-zone interfaces lt-0/0/0.0
set security zones security-zone Internet-VR-zone host-inbound-traffic system-services all
set security zones security-zone Internet-VR-zone host-inbound-traffic protocols all
set security zones security-zone Internet-VR-zone interfaces ge-0/0/3.0
set security zones security-zone Internet-VR-zone interfaces lt-0/0/0.1
set security policies default-policy permit-all
set routing-instances Internet-VR instance-type virtual-router interface lt-0/0/0.1
set routing-instances Internet-VR instance-type virtual-router interface ge-0/0/3.0
set protocols ospf area 0.0.0.0 interface ge-0/0/2.0
set protocols ospf area 0.0.0.0 interface lt-0/0/0.0
set routing-instances Internet-VR protocols ospf area 0.0.0.0 interface lt-0/0/0.1
set routing-instances Internet-VR protocols ospf area 0.0.0.0 interface ge-0/0/3.0
set firewall family inet filter bypass-flow-filter term bypass-flow-term then accept
set firewall family inet filter bypass-flow-filter term bypass-flow-term then packet-mode
set interfaces ge-0/0/2 unit 0 family inet bypass-flow-filter
set interfaces lt-0/0/0 unit 0 family inet bypass-flow-filter
```

```
{device R2}
set interfaces description “Connect to Internet-VR” ge-0/0/3 unit 0 family inet address 203.0.113.9/24
set protocols ospf area 0.0.0.0 interface ge-0/0/3
```

Step-by-Step Procedure

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

To configure selective stateless packet-based services for end-to-end packet-based forwarding:

1. Configure the IP addresses for the interfaces.

```
{device R0}
[edit]
user@host# set interfaces description “Connect to Master VR” ge-0/0/2 unit 0 family inet address
    198.51.100.9/24
```

```
{device R1}
[edit]
user@host# set interfaces description “Connect to R0” ge-0/0/2 unit 0 family inet address 198.51.100.10/24
user@host# set interfaces description “Connect to R2” ge-0/0/3 unit 0 family inet address 203.0.113.5/24
```

```
{device R2}
[edit]
user@host# set interfaces description “Connect to Internet-VR” ge-0/0/3 unit 0 family inet address
    203.0.113.9/24
```

2. Set an internal service interface between routing instances.

```
{device R1}
[edit]
user@host# set interfaces lt-0/0/0 unit 0 encapsulation frame-relay dlci 100 peer-unit 1 family inet address
    192.0.2.1/16
user@host# set interfaces lt-0/0/0 unit 1 encapsulation frame-relay dlci 100 peer-unit 0 family inet address
    192.0.2.2/16
```

3. Configure security zones.

```
{device R1}
[edit]
user@host# set security zones security-zone Master-VR-zone host-inbound-traffic system-services all
user@host# set security zones security-zone Master-VR-zone host-inbound-traffic protocols all
user@host# set security zones security-zone Master-VR-zone interfaces ge-0/0/2.0
user@host# set security zones security-zone Master-VR-zone interfaces lt-0/0/0.0
user@host# set security zones security-zone Internet-VR-zone host-inbound-traffic system-services all
user@host# set security zones security-zone Internet-VR-zone host-inbound-traffic protocols all
user@host# set security zones security-zone Internet-VR-zone interfaces ge-0/0/3.0
user@host# set security zones security-zone Internet-VR-zone interfaces lt-0/0/0.1
```

4. Configure policies.

```
{device R1}
[edit]
user@host# set security policies default-policy permit-all
```

5. Configure a virtual device routing instance.

```
{device R1}
[edit]
user@host# set routing-instances Internet-VR instance-type virtual-router interface lt-0/0/0.1
user@host# set routing-instances Internet-VR instance-type virtual-router interface ge-0/0/3.0
```

6. Enable OSPF on all interfaces in the network.

```
{device R0}
[edit]
user@host# set protocols ospf area 0.0.0.0 interface ge-0/0/2.0
```

```
{device R1 for Master-VR}
[edit]
user@host# set protocols ospf area 0.0.0.0 interface ge-0/0/2.0
user@host# set protocols ospf area 0.0.0.0 interface lt-0/0/0.0
```

```
{device R1 for Internet-VR}
[edit]
user@host#  set routing-instances Internet-VR protocols ospf area 0.0.0.0 interface lt-0/0/0.1
user@host#  set routing-instances Internet-VR protocols ospf area 0.0.0.0 interface ge-0/0/3.0
```

```
{device R2}
[edit]
user@host# set protocols ospf area 0.0.0.0 interface ge-0/0/3
```

7. Create a firewall filter and define a term for packet-based forwarding traffic.

```
{device R1}
[edit]
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term then accept
user@host# set firewall family inet filter bypass-flow-filter term bypass-flow-term then packet-mode
```

8. Apply the firewall filter to relevant interfaces.

```
{device R1}
[edit]
user@host# set interfaces ge-0/0/2 unit 0 family inet bypass-flow-filter
user@host# set interfaces lt-0/0/0 unit 0 family inet bypass-flow-filter
```

Results

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

```
{device R0}
[edit]
user@host# show interfaces
ge-0/0/2 {
    description “Connect to Master-VR”
    unit 0 {
        family inet {
            address 198.51.100.9/24
        }
    }
}
```

```
{device R0}
[edit]
user@host# show protocols
ospf {
    area 0.0.0.0/0 {
        interface ge-0/0/2.0;
    }
}
```

```
{device R2}
[edit]
user@host# show interfaces
ge-0/0/3 {
    description “Connect to Internet-VR”
    unit 0 {
        family inet {
            address 203.0.113.9/24;
        }
    }
}
```

```
{device R2}
[edit]
user@host# show protocols
ospf {
    area 0.0.0.0/0 {
        interface ge-0/0/3.0;
    }
}
```

```
{device R1}
[edit]
user@host# show interfaces
ge-0/0/2 {
    description “Connect to R0”
    unit 0 {
        family inet {
            filter {
                input bypass-flow-filter;
                }
            address 198.51.100.10/24;
        }
    }
}
lt-0/0/0 {
    unit 0 {
        encapsulation frame-relay;
        dlci 100;
        peer-unit 1;
        family inet {
            filter {
                input bypass-flow-filter
            }
            address 192.0.2.1/16;
        }
    }
    unit 1{
        encapsulation frame-relay;
        dlci 100;
        peer-unit 0;
        family inet {
            address 192.0.2.2/16 ;
        }
    }
}
{device R1}
[edit]
user@host# show protocols
ospf {
    area 0.0.0.0/0 {
        interface ge-0/0/2.0;
        interface lt-0/0/0.0;
    }
}
```

```
{device R1}
[edit]
user@host# show firewall
filter bypass-flow-filter {
    term bypass-flow-term {
        then {
            packet-mode;
            accept;
        }
    }
}
```

```
{device R1}
[edit]
user@host# show routing-instances
Internet-VR {
    instance-type virtual-router;
    interface lt-0/0/0.1;
    interface ge-0/0/3.0;
    protocols {
        ospf {
            area 0.0.0.0 {
                interface ge-0/0/3.0;
                lt-0/0/0.1;
            }
        }
    }
}
```

```
{device R1}
[edit]
user@host# show security
security zone Master-VR-zone {
    host-inbound-traffic {
        system-services {
            all;
            {
            protocols {
            all;
        {
    {
    intefaces {
    ge-0/0/2.0;
```

```
    lt-0/0/0.0;
    {
{
security zone Internet-VR-zone {
    host-inbound-traffic {
        system-services {
            all;
        {
        protocols {
            all;
        }
    }
    intefaces {
    ge-0/0/3.0;
    lt-0/0/0.1;
    {
{
policies {
    default-policy {
        permit-all;
    }
}
```

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

Verification

IN THIS SECTION

- Verifying the Packet-Based to Flow-Based Configuration | **237**
- Verifying Session Establishment on LAN Traffic | **238**
- Verifying Session Establishment on Internet Traffic | **239**

Confirm that the configuration is working properly.

Verifying the Packet-Based to Flow-Based Configuration

Purpose

Verify that the selective stateless packet-based services are configured for packet-based to flow-based forwarding.

Action

From configuration mode, enter the **show interfaces**, **show protocols**, **show security**, **show routing-instances**, and **show firewall** commands.

Verify that the output shows the intended configuration of the firewall filter, routing instances, interfaces, and policies.

Verify that the terms are listed in the order in which you want the packets to be tested. You can move terms within a firewall filter by using the **insert** command.

Verifying Session Establishment on LAN Traffic

Purpose

Verify that the sessions are established when traffic is transmitted on interfaces within the LAN.

Action

To verify that sessions are established, perform the following tasks:

1. On device **R1**, from operational mode enter the **clear security flow session all** command to clear all existing security flow sessions.

2. On device **R0**, from operational mode enter the **ping** command to transmit traffic to device **Master-VR**.

3. On device **R1**, with traffic transmitting from devices **R0** through **R1**, from operational mode enter the **show security flow session** command.

> **NOTE:** To verify established sessions, ensure that you enter the **show security flow session** command while the **ping** command is sending and receiving packets.

```
{device R0}
user@host> ping 192.0.2.1
```

```
PING 192.0.2.1 (192.0.2.1): 56 data bytes
64 bytes from 192.0.2.1: icmp_seq=0 ttl=63 time=2.208 ms
64 bytes from 192.0.2.1: icmp_seq=1 ttl=63 time=2.568 ms
64 bytes from 192.0.2.1: icmp_seq=2 ttl=63 time=2.573 ms
64 bytes from 192.0.2.1: icmp_seq=3 ttl=63 time=2.310 ms
64 bytes from 192.0.2.1: icmp_seq=4 ttl=63 time=1.566 ms
64 bytes from 192.0.2.1: icmp_seq=5 ttl=63 time=1.569 ms
...
```

```
{device R1}
user@host> show security flow session
```

```
0 sessions displayed
```

The output shows traffic transmitting from **R0** to **Master-VR** and no sessions are established. In this example, you applied the **bypass-flow-filter** with the **packet-mode** action modifier on interfaces **ge-0/0/0** and **lt-0/0/0.0** for your company's LAN traffic. This output verifies that the traffic between the two interfaces is correctly bypassing flow-based forwarding and hence no sessions are established.

Verifying Session Establishment on Internet Traffic

Purpose

Verify that sessions are established when traffic is transmitted to the Internet.

Action

To verify that traffic to the Internet is using flow-based forwarding and sessions are established, perform the following tasks:

1. On device **R1**, from operational mode enter the **clear security flow session all** command to clear all existing security flow sessions.

2. On device **R0**, from operational mode enter the **ping** command to transmit traffic to device **R2**.

3. On device **R1**, with traffic transmitting from **R0** to **R2** through **R1**, from operational mode enter the **show security flow session** command.

```
root@host> show security flow session
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

NOTE: To verify established sessions, ensure that you enter the **show security flow session** command while the **ping** command is sending and receiving packets.

```
{device R0}
user@host> ping 192.0.2.1 -c 10
```

```
PING 60.0.0.1 (60.0.0.1) 56(84) bytes of data.
64 bytes from 192.0.2.1: icmp_seq=1 ttl=64 time=1.98 ms
64 bytes from 192.0.2.1: icmp_seq=2 ttl=64 time=1.94 ms
64 bytes from 192.0.2.1: icmp_seq=3 ttl=64 time=1.92 ms
64 bytes from 192.0.2.1: icmp_seq=4 ttl=64 time=1.89 ms

...
```

```
{device R1}
user@host> show security flow session
```

```
Session ID: 189900, Policy name: default-policy/2, Timeout: 2
  In: 198.51.100.9/0 --> 192.0.2.1/5924;icmp, If: lt-0/0/0.1
  Out: 192.0.2.1/5924 --> 198.51.100.9/0;icmp, If: ge-0/0/3.0

Session ID: 189901, Policy name: default-policy/2, Timeout: 2
  In: 198.51.100.9/1 --> 192.0.2.1/5924;icmp, If: lt-0/0/0.1
  Out: 192.0.2.1/5924 --> 198.51.100.9/1;icmp, If: ge-0/0/3.0

Session ID: 189902, Policy name: default-policy/2, Timeout: 4
  In: 198.51.100.9/2 --> 192.0.2.1/5924;icmp, If: lt-0/0/0.1
  Out: 192.0.2.1/5924 --> 198.51.100.9/2;icmp, If: ge-0/0/3.0

 3 sessions displayed
```

The output shows traffic transmitting from devices **R0** to **R2** and established sessions. In this example, you did not apply the **bypass-flow-filter** with the **packet-mode** action modifier on routing instance **Internet-VR** for your company's Internet traffic. This output verifies that the traffic to the Internet is correctly using flow-based forwarding and hence sessions are established.

Note that sessions are established only when traffic is flowing between **lt-0/0/0.1** and **ge-0/0/3** and not when traffic is flowing between **ge-0/0/2** and **lt-0/0/0.0**.

Understanding Session Cache

IN THIS SECTION

- Overview | 241
- Selective Session Cache Installation | 242
- IPsec VPN Session Affinity Enhancement Using Session Cache | 243
- Fragmentation Packet Ordering Using NP Session Cache | 244

Overview

The SRX5K-MPC (IOC2), SRX5K-MPC3-100G10G (IOC3), and SRX5K-MPC3-40G10G (IOC3) on SRX5400, SRX5600, and SRX5800 devices support session cache and selective installation of the session cache.

Session cache is used to cache a conversation between the network processor (NP) and the SPU on an IOC. A conversation could be a session, GTP-U tunnel traffic, IPsec VPN tunnel traffic, and so on. A conversation has two session cache entries, one for incoming traffic and the other for reverse traffic. Depending on where the traffic ingress and egress ports are, two entries might reside in the same network processor or in different network processors. IOCs support session cache for IPv6 sessions.

A session cache entry is also called a *session wing*.

Session cache on the IOC leverages Express Path (formerly known as *services offloading*) functionality and helps prevent issues such as high latency and IPsec performance drop.

A session cache entry records:

- To which SPU the traffic of the conversion should be forwarded
- To which egress port the traffic of the conversion should be forwarded in Express Path mode
- What processing to do for egress traffic, for example, NAT translation in Express Path mode

Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the session cache of the sessions in the IOC helps to solve certain performance issues. The SPU can now instruct the IOC session cache to forward subsequent traffic to a specific anchor SPU.

Starting with Junos OS Release 15.1X49-D10, the SRX5K-MPC (IOC2) and the IOC3 support VPN session affinity through improved flow module and session cache. Starting in Junos OS Release 12.3X48-D30, on the IOC2, VPN session affinity through session cache is supported.

Other traffic was hashed to SPUs based on their 5-tuple key information. VPN traffic employed the concept of the anchored SPU, which did not necessarily coincide with the functions of the flow SPU. The network

processor could only forward the packets to the flow SPU based on the 5-tuple hash. The flow SPU then forwarded the packet to the anchored SPU. This created an extra hop for VPN traffic, which wasted the switch fabric bandwidth and reduced the VPN throughput roughly by half. This performance reduction occurred because the traffic still had to go back to the flow SPU after processing on the anchored SPU.

The session cache table is now extended on IOC to support the NP sessions. Express Path traffic and NP traffic share the same session cache table on IOCs. Express Path traffic is forwarded by the IOC itself either locally or to another IOC, because the traffic does not require any services from the SPU. NP traffic is forwarded to the SPU specified in the session cache for further processing. All the session cache entries are shared by both Express Path session traffic and NP traffic.

To enable session cache on the IOCs you need to run the **set chassis fpc *<fpc-slot>* np-cache** command.

> **NOTE:** The IOC2 and the IOC3 utilize the delay sessions delete mechanism. The same sessions (sessions with the same five tuples) that are deleted and then reinstalled immediately are not cached on the IOCs.

Selective Session Cache Installation

To avoid high latency, improve IPSec performance, and to better utilize the valuable resources, certain priority mechanisms are applied to both flow module and the IOC.

The IOCs maintain and monitor session cache usage threshold levels. The IOCs also communicate the session cache usage to the SPU, so that when a certain session cache usage threshold is reached, the SPU only sends session cache installation requests for selective high-priority traffic sessions.

Applications like IDP, ALG need to process packets in order. One SPU has multiple flow threads to handle packets belong to one session, LBT, POT packet order can make sure traffic pass through firewall in order, it cannot guarantee application to process packets that belong to same session in order. Flow serialization provides the method that only one SPU flow thread processing packets belong to the same session at one time, so applications can receive, process and send out packet in order. Other flow threads can do flow serialization processing for other sessions at the same time.

The following four priority levels are used to determine which type of traffic can install session cache on the IOCs:

- **Priority 1 (P1)**— IPSec and Express Path qualified traffic
- **Priority 2 (P2)**— Fragmentation ordering
- **Priority 3 (P3)**— NAT/SZ (Session serialization) traffic traffic
- **Priority 4(P3)**— All other types of traffic

The IOCs maintain and monitor the threshold levels for session cache usage and update the current real-time session cache usage to the SPU. The SPU requests the IOC to install the session cache for certain high-priority traffic sessions. Session cache usage for high-priority traffic sessions is defined in table:

Table 16: Session Cache Installation Bars

Traffic Type	0% < utilization < 25%	25% < utilization < 50%	50% < utilization < 75%	75% < utilization < 100%
IPsec and Express Path traffic	Yes	Yes	Yes	Yes
Fragmentation Ordering traffic	Yes	Yes	Yes	No
NAT/SZ traffic	Yes	Yes	No	No
Other traffic	Yes	No	No	No

To conserve session entries on the IOC, the flow module selectively installs sessions on the IOC. To facilitate the session install selection, the IOC maintains corresponding thresholds to provide an indication to the flow module (on how full the session cache table is on the IOCs).Two bits in the meta header are added to indicate the current cache table utilization status. All packets going to the SPU will carry these two status bits to inform the flow module of the utilization of the cache table on the IOC.

IPsec VPN Session Affinity Enhancement Using Session Cache

SRX Series devices are fully distributed systems, and an IPsec tunnel is allocated and anchored to a specific SPU. All the traffic that belongs to an IPsec tunnel is encrypted and decrypted on its tunnel-anchored SPU. In order to achieve better IPsec performance, IOC improves the flow module to create sessions for IPsec tunnel-based traffic (before encryption and after decryption) on its tunnel-anchored SPU, and installs session cache for the sessions so that the IOC can redirect the packets directly to the same SPU to minimize packet-forwarding overhead. Express Path traffic and NP traffic share the same session cache table on IOCs.

You need to enable session cache on the IOCs and set the security policy to determine whether a session is for Express Path (formerly known as *services offloading*) mode on the selected Flexible PIC Concentrator (FPC).

To enable IPsec VPN affinity use, the **set security flow load-distribution session-affinity ipsec** command.

> **NOTE:** To enable IPsec VPN affinity, you must also enable the session cache on IOCs by using the **set chassis fpc *<fpc-slot>* np-cache** command.

Fragmentation Packet Ordering Using NP Session Cache

A session might consist of both normal and fragmented packets. With hash-based distribution, 5-tuple and 3-tuple key can be used to distribute normal and fragmented packets to different SPUs, respectively. On SRX Series devices, all the packets of the session are forwarded to a processing SPU. Due to forwarding and processing latency, the processing SPU might not guarantee packet ordering of the session.

Session cache on the IOCs ensure ordering of packets of a session with fragmented packets. A session cache entry is allocated for normal packets of the session and a 3-tuple key is used to find the fragmented packets. On receipt of the first fragmented packet of the session, the flow module allows the IOC to update the session cache entry to remember the fragmented packets for the SPU. Later, IOC forwards all subsequent packets of the session to the SPU to ensure ordering of packets of a session with fragmented packets.

Understanding Symmetric Fat IPsec Tunnel

Starting from Junos OS 19.4R1, on SRX5400, SRX5600, SRX5800, and vSRX instances, fat tunnel technology is introduced to improve the a IPsec tunnel throughput value up to 10 times of current value.

A new CLI command is introduced to enable the fat IPsec tunnel. The fat IPsec tunnel feature is disabled by default. The new CLI command introduced is **fat-core** in the **set security distribution-profile** hierarchy. When you enable the fat-core, the below configuration is displayed:

```
security {
    distribution-profile {
        fat-core;
    }
}
```

Before configuring the fat IPsec tunnel, make sure the following are configured.

- For fast path forwarding, configure the IOC cache for the session information using the **set chassis fpc *FPC slot* np-cache** command.
- To enable session affinity, use the **set security flow load-distribution session-affinity ipsec** command.
- To enable Power mode, use the **set security flow power-mode-ipsec** command.

Release History Table

Release	Description
19.4R1	Starting from Junos OS 19.4R1, on SRX5400, SRX5600, SRX5800, and vSRX instances, fat tunnel technology is introduced to improve the a IPsec tunnel throughput value up to 10 times of current value.
15.1X49-D30	Starting in Junos OS Release 15.1X49-D30 and Junos OS Release 17.3R1, the session flow summaries include CP session IDs.
15.1X49-D10	Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the session cache of the sessions in the IOC helps to solve certain performance issues.
15.1X49-D10	Starting with Junos OS Release 15.1X49-D10, the SRX5K-MPC (IOC2) and the IOC3 support VPN session affinity through improved flow module and session cache
12.1X48-D30	Starting in Junos OS Release 12.3X48-D30, on the IOC2, VPN session affinity through session cache is supported

RELATED DOCUMENTATION

Express Path | 249

Reverse Route Packet Mode using Virtual Router

During flow processing, when the traffic route between the server and client is changed, the traffic reroutes using the virtual router (VR). The VR used in rerouting is available in the interface or the filter-based forwarding (FBF). The behavior of the reroute is monitored using the **set security flow advanced-options reverse-route-packet-mode-vr** command.

> **NOTE:** The **reverse-route-packet-mode-vr** command works on root logical system and is enabled globally.

When the reverse route option is enabled, there is no change in the packet flow. When the reverse route option is disabled, the route lookup uses the VR from the packet incoming interface. If the VR in the route is incorrectly configured, then the traffic between the server and client is dropped.

> **NOTE:** The resolve reserve route in the flow first path is not configured as the VR information from the client to the server packet is not available.

For example, Figure 27 on page 246 shows the behavior of the packet flow when the **reverse-route-packet-mode-vr** command is not configured. The client to server traffic uses the routing instance VR2 of incoming interface ge-0/0/0.0 to route the traffic. The server to client traffic also uses the routing instance VR2 of incoming interface ge-0/0/0.0 to route the traffic.

Figure 27: Reverse Route Disabled

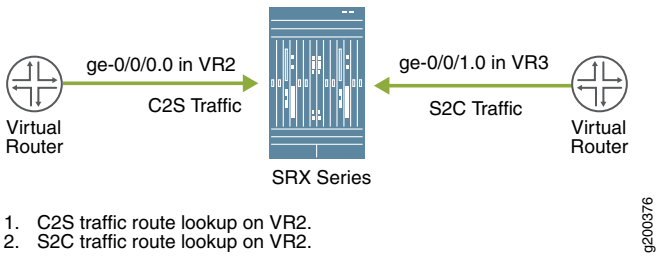


Figure 28 on page 247 shows the behavior of the packet flow when the **reverse-route-packet-mode-vr** command is configured using interface. The client to server traffic uses the routing instance VR2 of incoming interface ge-0/0/0.0 to route the traffic. The server to client traffic uses the routing instance VR3 of interface ge-0/0/1.0 to route the traffic.

Figure 28: Reverse Route Enabled with Interface

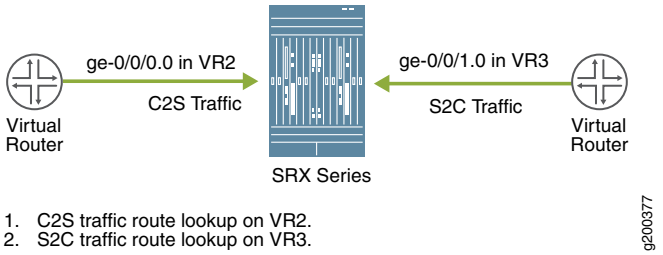
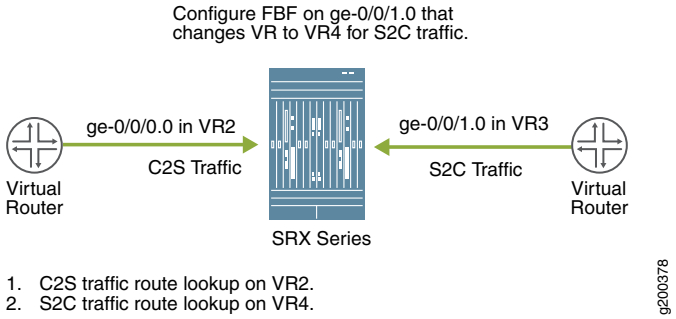


Figure 29 on page 247 shows the behavior of the packet flow when the **reverse-route-packet-mode-vr** command is configured using FBF. The client to server traffic uses the packet incoming interface ge-0/0/0.0 in VR2 to route the traffic. Configuring FBF on the interface ge-0/0/1.0 changes VR3 to VR4. The server to client traffic uses VR4 to route the traffic.

Figure 29: Reverse Route Enabled with FBF



Understanding To-host Traffic on Virtual Router

On a SRX Series device, all the traffic that passes the firewall filter is referred as to-host traffic. The traffic from the firewall to the device is referred as the from-host traffic. The to-host traffic uses an egress interface and the from-host traffic uses an ingress interface. If both the interfaces are not in the same routing instance, there will be a session mismatched. To overcome this issue, the to-host and the from-host traffic choose interfaces that are available in the same routing instance.

Figure 30 on page 248 shows the to-host traffic using the routing instance VR5 of interface ge-0/0/0.0 and the routing instance VR6 of destination interface lo0.1.

Figure 30: To-host Traffic on VR

For example, if the to-host traffic uses a local interface (such as local....X) that is in routing instance 5 (VR5), and the from-host traffic uses the interface in routing instance 6 (VR6). The session output displaying the interface information of the to-host traffic is:

```
Session ID: 10000179, Policy name: pol1/4, Timeout: 2, Valid
  In: 192.168.90.1/4 --> 192.168.91.1/19050;icmp, Conn Tag: 0x0, If: xe-9/0/3.0,
Pkts: 1, Bytes: 84, CP Session ID: 10000178
  Out: 192.168.91.1/19050 --> 192.168.90.1/4;icmp, Conn Tag: 0x0, If: .local..5,
Pkts: 1, Bytes: 84, CP Session ID: 10000178
```

NOTE: The session output displays the local interface of the to-host traffic as local....5.

To synchronize the to-host and from-host traffics, the to-host traffic uses traffic destination IP interface (lo0.1) that is available in VR6. As the from-host traffic is using the interface available in VR6, the session matches. The session output displaying the interface information of the to-host traffic is:

```
Session ID: 10000179, Policy name: pol1/4, Timeout: 2, Valid
  In: 192.168.90.1/4 --> 192.168.91.1/19050;icmp, Conn Tag: 0x0, If: xe-9/0/3.0,
Pkts: 1, Bytes: 84, CP Session ID: 10000178
  Out: 192.168.91.1/19050 --> 192.168.90.1/4;icmp, Conn Tag: 0x0, If: .local..6,
Pkts: 1, Bytes: 84, CP Session ID: 10000178
```

NOTE: The session output displays the local interface of the to-host traffic as local....6.

RELATED DOCUMENTATION

Express Path

IN THIS SECTION

- Express Path Overview | 249
- Understanding the Express Path Solution | 265
- Enabling and Disabling Express Path | 266
- Example: Enabling Express Path in Security Policies | 267
- Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path | 269
- Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path | 272
- Example: Configuring SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) on an SRX5000 Line Device to Support Express Path | 275
- Example: Configuring Express Path on an SRX5000 Line Device with IOC3 | 278
- Example: Configuring Low Latency | 284
- Managing Packet Fragmentation in IPsec VPN Networks | 287

Express Path (formerly known as *services offloading*) is a mechanism for processing fast-path packets in the network processor instead of in the Services Processing Unit (SPU). This method reduces the packet-processing latency that arises when packets are forwarded from network processors to SPUs for processing and back to I/O cards (IOCs) for transmission.

Express Path Overview

IN THIS SECTION

- Understanding Express Path Functionality | 250
- Understanding Express Path Support on SRX Series Devices | 251
- Understanding Express Path Features | 252
- Express Path Limitations | 257

- Express Path Support on NP-IOC Card | **259**
- Express Path Support on SRX5K Modular Port Concentrator | **259**
- Express Path Support on SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) | **260**
- IPv6 Flow in Express Path Mode for IOC2 and IOC3 | **263**
- IPv6 Flow in Express Path Mode | **264**

Understanding Express Path Functionality

This feature is supported on SRX4600, SRX5400, SRX5600, and SRX5800 devices.

Express Path considerably reduces packet-processing latency by 500–600 percent.

When the first packet arrives at the interface, the network processor forwards it to the SPU. If the SPU verifies that the traffic is qualified for Express Path, an Express Path session is created on the network processor. If the traffic does not qualify for Express Path, a normal session is created on the network processor. If an Express Path session is created, the subsequent fast-path packets are processed in the network processor itself.

> **NOTE:** A normal session forwards packets from the network processor to the SPU for fast-path processing, whereas an Express Path session processes fast-path packets in the network processor and the packets exit out of the network processor itself.

When an Express Path session is created on the network processor, subsequent packets of the flow match the session on the network processors. The network processor then processes and forwards the packet. The network processor also handles additional processing such as TCP sequence check, time to live (TTL) processing, Network Address Translation (NAT), and Layer 2 header translation.

The network processor forwards packets to the SPU in the following cases:

- When the first packet arrives at the interface, the network processor forwards it to the central point (CP). The central point in turn forwards the packet to the SPU. The SPU then creates a session on the network processor.
- When an SPU session exists even if no network processor session exists, the network processor forwards a packet to the central point, which in turn forwards the packet to the SPU. The SPU then creates a session on the network processor.
- When a packet matches a normal session on the network processor, it is forwarded to the SPU.

Starting with Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, a license is no longer required to enable Express Path functionality. Your previously acquired license will not be effective anymore. (Prior to Junos OS Release 12.3X48-D10, Express Path was a licensed software feature.)

Starting with Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, SRX5400, SRX5600, and SRX5800 devices with the SRX5K-MPC (IOC2), SRX5K-MPC3-100G10G (IOC3), and SRX5K-MPC3-40G10G (IOC3) support Express Path (formerly known as services offloading) for ALG traffic.

The following ALG data traffic that supports Express Path—FTP, H.323 (only RTP/RTCP sessions are offloaded), MGCP, MS RPC, RSH, RTSP, SCCP, SIP (only RTP/RTCP sessions are offloaded), SUN RPC, TALK (only TCP sessions are offloaded), and TFTP.

DNS, IKE and ESP, PPTP, and SQL-NET ALG data traffic do not support Express Path.

Once an Express Path session is setup, packets cannot be sent to the SPU again.

Understanding Express Path Support on SRX Series Devices

Table 17 on page 251 provides details about the Express Path support on different SRX Series cards.

Starting from Junos OS Release 19.3R1, Express Path is supported on SRX4600 platform and enabled by default. There is no CLI command to enable or disable this feature.

Table 17: Express Path Support on SRX Series Device Cards

SRX Series Device	Card Name and Model Number	Earliest Supported Release
SRX5000 Line Devices I/O Cards (IOCs)		
SRX5600, SRX5800	SRX5K-40GE-SFP	Junos OS Release 11.4
SRX5600, SRX5800	SRX5K-4XGE-XFP	Junos OS Release 11.4
SRX5600, SRX5800	SRX5K-FPC-IOC containing one of the following cards: • SRX-IOC-16GE-TX • SRX-IOC-4XGE-XFP • SRX-IOC-16GE-SFP	Junos OS Release 11.4
SRX5400, SRX5600, SRX5800	SRX5K-MPC containing one of the following MICs: • SRX-MIC-10XGE-SFFP • SRX-MIC-2X40GE-OSFP • SRX-MIC-1X100GE-CFP • SRX-MIC-20GE-SFP	Junos OS Release 12.3X48-D10

Table 17: Express Path Support on SRX Series Device Cards *(continued)*

SRX Series Device	Card Name and Model Number	Earliest Supported Release
SRX5400, SRX5600, SRX5800	SRX5K-MPC3 (IOC3) containing one of the following MPCs: • SRX5K-MPC3-40G10G (24x10GE + 6x40GE MPC) • SRX5K-MPC3-100G10G (2x100GE + 4x10GE MPC)	Junos OS Release 15.1X49-D10
SRX4600	NA	Junos OS Release 19.3R1

Different Express Path features are supported on different cards for different Junos OS releases. See the *Junos OS Release Notes* for details.

NOTE: On SRX Series devices with chassis cluster, the np-cache must be enabled only on the FPC. The local node configuration is enough for both the nodes in a chassis cluster.

NOTE: On the SRX5600 and SRX5800 Services Gateways, the Express Path sessions for traffic that traverse between legacy IOC cards and the SRX5K-MPC or the SRX5K-MPC3 are not supported.

The Express Path sessions traversing only on legacy IOC cards or only on the SRX5K-MPC (IOC2), the SRX5K-MPC3-100G10G (IOC3), or the SRX5K-MPC3-40G10G (IOC3) are supported. However, the SRX5K-MPC (IOC2), the SRX5K-MPC3-100G10G (IOC3), the SRX5K-MPC3-40G10G (IOC3), and the legacy IOCs can still be present on the same chassis.

Understanding Express Path Features

Wing Statistics Counter

The network processor in Express Path mode provides the option for each flow entry to keep a per-wing bytes counter. The counter captures the number of bytes that the network processor sends out over the wing.

When the counter is enabled, for every ingress packet, the network processor searches its flow entry (a session wing). If the packet belongs to an established flow entry, the network processor increases the byte counter of the flow entry by byte count in the packet. The network processor periodically copies a packet (called a copy-packet) of each flow entry to its associated SPU, allowing the SPU to maintain the session.

The network processor sends flow byte counter values in the header of copy-packet packets. The SPU accumulates and keeps per-wing statistics counters.

> **NOTE:** The counter value carried to the SPU is always one packet short to allow the SPU to add the current packet's byte count to the counter to get the correct total. For example, if packet N's copy carries a counter value to the SPU, the counter value is the total bytes received in the flow up to packet N-1.

The counter value does not include packets that were sent before the session was set up on the network processor. Therefore, the SPU might need to account for the three-way handshake packet and other packets sent through the SPU. The actual session byte counter shown on the SPU might be short by the amount of bytes sent by the client during the copy interval. This discrepancy results because these bytes can be counted locally by the network processor, but have not yet been reported to the SPU.

> **NOTE:** You cannot change the statistics configuration during the life cycle of a live session. Disabling or enabling the per-wing statistics configuration while a session is alive at the network processor invalidates the session statistics on the current session. The new sessions statistics can be valid only after the configuration changes are committed . Network processor per-wing counters cannot be cleared.

> **NOTE:** Wing statistics counter configuration is enabled, by default, on SRX5400, SRX5600, and SRX5800 devices with the SRX5K-MPC (IOC2) and the SRX 5K-MPC3 (IOC3).

Sessions per Wing Statistics

The NP-IOC has a larger static RAM (SRAM) to accommodate session resources, thus hosting more sessions per PIC. Table 18 on page 253 displays the total number of session wings, including both Express Path and non-Express Path.

Table 18: Total Number of Sessions per Wing in Network Processor Express Path Configuration Mode

Total Number of Wings		Number of Express Path UDP Wings		Number of Express Path TCP Wings	
Cards and SRX Series Device	**Non-Express Path Mode Sessions**	**Without Statistics**	**With Statistics**	**Without Statistics**	**With Statistics**
IOC	1.3 million	1.3 million	900,000	600,000	400,000

Table 18: Total Number of Sessions per Wing in Network Processor Express Path Configuration Mode *(continued)*

Total Number of Wings		Number of Express Path UDP Wings		Number of Express Path TCP Wings	
FIOC	2.3 million	198,000	900,000	600,000	400,000
SRX5000 line device SRX5K-MPC	NA	1.8 million	1.8 million	1.8 million	1.8 million
SRX5000 line device SRX5K-MPC3 (IOC3)	NA	2.0 million	2.0 million	2.0 million	2.0 million
SRX5000 line device SRX5K IOC4	NA	2.0 million	2.0 million	2.0 million	2.0 million
SRX4600	NA	2.0 million	2.0 million	2.0 million	2.0 million

Cross-Network Traffic

Express Path provides additional cross-network-processor support; therefore, it is no longer restricted to the ports of the same network processor. If network processors for both the ingress and egress ports are in Express Path mode, then Express Path packets are directly forwarded from the ingress network processor to the egress network processor in the fast-flow path. Packets cross switch fabric when they are forwarded from one network processor to another, thus increasing the latency of the packet. In Express Path mode, the latency of cross-network-processor packets is higher than the packets that are forwarded within an individual network processor.

NOTE: The SRX5K-MPC receives session messages from the SPU. The session messages carry the information to support inter- and intra-Packet Forwarding Engine Express Path for IPv4.

LAG Support in Express Path Mode

Ethernet link aggregation groups (LAGs) combine links and provide increased bandwidth and link availability. Express Path reduces packet latency by processing and forwarding packets in the network processor instead of in the Services Processing Unit (SPU). Supporting LAG in the Express Path mode combines the benefits of both these features and provides enhanced throughput, link redundancy, and reduced packet latency.

LAG Links Qualifying for Express Path Mode

You can use the links in a LAG as ingress or egress interfaces in Express Path mode. The LAG links can include links from different network processors (in case of legacy cards such as IOCs or Flex IOCs) or from the same modular port concentrator (in case of SRX5K-MPC). For a LAG link to qualify for Express Path, all its member links should be connected to Express Path-enabled network processors. If Express Path is disabled on any of the member links in a LAG, a regular session (non-Express Path session) is created. Also, LAG links are not supported between legacy cards such as IOCs or Flex IOCs and the SRX5K-MPC.

LAG and Network Processor Wings

The network processor checks the egress interface in the Express Path mode for each wing. Per-wing traffic distribution over a LAG interface is achieved by letting the SPU install wings pointing to egress interfaces with a balanced distribution.

The network processor periodically copies a packet (called a *copy-packet*) to the SPU, allowing it to maintain the session. The copy-packet contains the egress interface information, which the SPU uses to handle LAG member change cases; for example, when a link is down or disabled. When there is no member interface that can be used as an egress interface for transmitting traffic, the network processor session is updated from Express Path to non-Express Path and the packet is sent to the SPU. A new Express Path network processor session is then installed using a new, valid egress interface.

- First wing—On the egress interface, the SPU selects one LAG active member link as the outgoing interface for a specific fast-forward session. The SPU treats this active member interface just like any physical interface in the Express Path mode and records the interface to the network processor fast-forward session. After that, all traffic that matches this network processor session is directly transmitted through that member link.
- Reverse wing—the reverse network processor session is installed only when all LAG member links are connected to a single network processor. When member links of a LAG are from multiple network processors, the reverse network processor session is initiated by reverse traffic later (it is not preinstalled).

  The Express Path network processor session needs to have outgoing interface information to send traffic. If the incoming interface is a physical interface, then it can be used as an outgoing interface for the reverse wing. However, if the incoming interface is an aggregated Ethernet interface, the SPU selects a member interface to be the outgoing interface.

LAG and Network Processor Session Updates

Some changes in the LAG interfaces can cause network processor session updates:

- LAG interface status changes—The LAG interface status can change due to several reasons—for example, when member interfaces are deleted, when an interface is down, or when an active LAG member is removed. In such cases, a session scan is triggered and network processor sessions related to the LAG interface are removed. The packet then installs a new network processor session, which could be a fast-forward session, if it qualifies.

- An active member is deactivated or removed from the LAG—In cases when the LAG still has an active member, neither a reroute nor session scan is triggered. The traffic is redistributed on the failed LAG member by monitoring outgoing logical interface status in the SPU.
- A new member is added to the LAG—The network processor session is not updated. A new network processor session is created, which might use the newly added interface or not, depending on the member selection algorithm for the LAG.

Redistribution of Traffic on a Failed LAG Interface

When a LAG member fails, the traffic needs to be redistributed. To redistribute traffic, the system monitors the status of the egress interface in the SPU. When the system detects a failure, it updates the Routing Engine kernel, and passes the physical interface information down to all SPUs. On receiving a copy of the session, the SPU extracts the egress interface index and checks the physical interface information. If the physical interface is down, the SPU uninstalls the session and the ingress network processor deletes the session cache.

For the next ingress packet of the same conversation, the network processor forwards the packet to the SPU to select an active member interface in the LAG as an egress interface. The SPU performs the distribution algorithm to select a new egress interface. A new session with the new egress interface index is installed in the ingress network processor and a new fast-flow path is created.

End-to-End Debugging

For regular flow packets, end-to-end debugging functions are the same as in the non-Express Path mode; packet filter and action items are supported in this flow mode. For traffic that matches Express Path sessions, the end-to-end debugging function supports one packet copy to the host CPU when the filter and the action are both affirmative in the end-to-end search results.

> **NOTE:** End-to-end debugging is not supported on the SRX5K-MPC when Express Path mode is enabled.

Per-Session Statistics CLI

To enable the per-session statistics, copy and paste the following command into the CLI at the **[edit]** hierarchy level.

```
set chassis fpc <fpc-slot> pic <pic_slot> services-offload per-session-statistics
```

Verify that the **services-offload per-session-statistics** command is enabled.

show configuration chassis

```
user@host> show configuration chassis
fpc 1 {
    pic 1 {
        services-offload {
            per-session-statistics;
        }
    }
 }
```

> **NOTE:** The **services-offload per-session-statistics** command is not applicable for the SRX5K modular port concentrators when Express Path is configured, because every session has statistics by default.

Use the **show chassis hardware** command to display hardware information.

show chassis fpc pic-status (SRX5600 and SRX5800 devices When Express Path [Services-Offload] is Configured)

`user@host>` **show chassis fpc pic-status**

```
Slot 0 Online SRX5k DPC 40x 1GE
PIC 0 Online 10x 1GE RichQ
PIC 1 Online 10x 1GE RichQ
PIC 2 Online 10x 1GE RichQ
PIC 3 Online 10x 1GE RichQ
Slot 2 Online SRX5k IOC II
PIC 0 Online 12x 10GE SFP+- np-cache/services-offload
Slot 3 Online SRX5k IOC II
PIC 0 Online 10x 10GE SFP+- np-cache/services-offload
PIC 2 Online 10x 10GE SFP+- np-cache/services-offload
Slot 5 Online SRX5k SPC
PIC 0 Online SPU Cp-Flow
PIC 1 Online SPU Flow
```

Express Path Limitations

The Junos OS Express Path implementation has the following limitations.

- Unsupported features—The following features are not supported with Express Path:

- Transparent mode is not supported. If transparent mode is configured, a normal (non-Express Path) session is installed.
- Only multicast sessions with one fan-out are supported. If a multicast session with more than one fan-out exists, a normal session is installed.
- Only active/passive chassis cluster configuration is supported. Active/active chassis cluster configuration is not supported.
- Fragmented packets are not supported. If fragmented packets exist, a normal session is installed.
- Express Path is not supported in IPsec VPN, and IDP configurations. Normal flow sessions will be used in these scenarios.
- Express Path does not support cross logical system traffic, regular flow sessions are used for cross logical system traffic processing and forwarding.
- Starting in Junos OS 15.1X49-D40 and Junos OS Release 17.3R1, IPv6 is supported. Prior to Junos OS 15.1X49-D40, IPv6 support is limited, and if IPv6 is configured, a normal session is installed.
- When Express Path mode is enabled on an SRX5K-MPC, you might not be able to enable the firewall filter. In general, all processes related to J-Flow (versions 5, 8, and 9) in an SRX Series device take place in the SPU. In Express Path security flow sessions, the J-Flow configuration will not take effect.
- Class of service (CoS) on egress interfaces is not supported.
- Configuration of protection against a teardrop (Screen) attack is not supported when Express Path is enabled.
- Configuring different MTU size values is not supported on the SRX5K-MPC when Express Path is enabled.

- Performance drop—The following drops in performance occur when Express Path is enabled:
  - Normal (non-Express Path) sessions—When Express Path is enabled, for normal sessions, the performance can drop by approximately 20 percent for connections per second (CPS) and 15 percent for packets per second (pps) when compared with normal sessions.
  - Express Path sessions—When Express Path is enabled, for fast-forward sessions, the performance can drop by approximately 13 percent for connections per second (CPS).
- Chassis cluster—When the device is operating in chassis cluster mode:
  - Asymmetric IOC configuration is not supported when Express Path is enabled on a device operating in chassis cluster mode.
  - If a child link goes down from the LACP-enabled redundant Ethernet interface of an IOC with Express Path enabled on its FPC, all traffic on this link is distributed to other active child links of the interface. If the child link comes up and rejoins the redundant Ethernet interface, then the existing traffic or sessions might not be redistributed over this newly rejoined active child link. New sessions might however traverse through this link.

  - If a new child link is added on the LACP-enabled redundant Ethernet interface of an IOC with Express Path enabled on its FPC, then the existing traffic or sessions might not be redistributed over this new child link. New sessions might however traverse through this link.

- For the normal flow sessions, the **show security flow session** command displays bytes counters based on IP header length. However for sessions in Express Path mode, the statistics is collected from IOC2 and IOC3 ASIC hardware engine, and includes full packet length with L2 headers. So the **show security flow session** command output displays slightly larger bytes counters for sessions in Express Path mode than the normal flow session.

Express Path Support on NP-IOC Card

The NP-IOC card integrates an existing I/O card (IOC) with a Network Processing Card (NPC) in one card with simplified Layer 2 functions in the hardware. This new hardware changes the way the interface is interpreted in the system.

> **NOTE:** Each interface in the NP-IOC card can only be attached to the network processor on the NP-IOC card. This fixed attachment setup requires the network processor to manage the interfaces as local or relative interfaces, instead of systemwide global interfaces.

Besides providing physical layer network connections, another function of the NP-IOC card is to distribute packets coming into the physical ports to the Services Processing Units (SPUs) and to forward packets out of the physical ports. For parallel security processing, flow sessions are assigned to multiple SPUs, based on a load balance algorithm. The network processor on the NP-IOC is responsible for directing traffic to the proper SPU based on the session table installed in its local memory.

In Express Path mode, the first packet is processed as is, meaning the packet is forwarded to the central point and the central point assigns an SPU and passes the packet to the SPU. For packets in fast-path, instead of forwarding all packets to the SPU, the network processor forwards the packets to an egress network processor, which can be different from or the same as the ingress network processor.

Express Path Support on SRX5K Modular Port Concentrator

The SRX5K-MPC is a Modular Port Concentrator (MPC) that is supported on the SRX5400, SRX5600, and SRX5800.

The SRX5K-MPC is an interface card with two slots that accept MICs, which add Ethernet ports to your services gateway. An MPC with MICs installed functions in the same way as a regular I/O card (IOC) but allows you to add different types of Ethernet ports to your device.

Each MPC is equipped with Trio chipsets, which perform control functions tailored to the MPC's media type.

When a Trio chipset receives the first packet, the packet is forwarded to an SPU based on the hash value (which is determined by a hash function of the 5 tuples of the session).

If the SPU verifies that the traffic is qualified for Express Path (formerly known as *services offload*), an Express Path session is created on the Trio chipset. If the traffic does not qualify for Express Path, it is forwarded by default hash-based forwarding to SPUs. If an Express Path session is created, the subsequent fast-path packets are processed in the Trio chipset itself.

The Trio chipset performs all the necessary checks to forward the packet, including TTL checking and decreasing, TCP sequence check, NAT translation, and Layer 2 header encapsulation. In addition, the Trio chipset sends a session refresh message to the SPU every second. This message is used to refresh the SPU session, detect the current state of the Trio chip set and update SPU session statistics.

The session table on the SRX5K-MPC, managed by the SPU, provides the following functions:

- Flow insert or delete
- Flow lookup
- Flow aging
- Flow statistics

The SPU inserts and deletes flow entries in the session table based on policy matching results.

> **NOTE:** Configuring the screen options on an SRX5K-MPC when operating in Express Path mode is the same as when the card is operating in normal mode.

Express Path Support on SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3)

Express Path (formerly known as *services offload*) on the IOC3 is based on processing fast-path packets through the Trio chipset instead of in the SPU to offload some basic firewall functions to the IOC3.

When the Express Path feature is enabled, the IOC3 provides much lower latency, and also supports higher throughput by removing the overload on the SPU. The IOC3 supports both intra-card traffic flow and inter-card traffic flow. To achieve the best latency results, both the ingress port and egress port of a traffic flow need to be on the same XM chip of the IOC3.

Starting with Junos OS Release 15.1X49-D80, two new system log messages have been added to indicate memory-related problems on the interfaces to the DDR3 memory.

These system log messages are:

- XMCHIP_CMERROR_DDRIF_INT_REG_CHKSUM_ERR_MINOR
- XMCHIP_CMERROR_DDRIF_INT_REG_CHKSUM_ERR_MAJOR

The error messages indicate that the XMCHIP on an Flexible PIC Concentrator (FPC) has detected a checksum error, which is causing packet drops. The following error threshold values classify the error as a major error or a minor error:

- Minor error —> 5 errors per second
- Major error —> 255 errors per second (maximum count)

The flow table on the IOC3 is managed by the SPU of the flow module. The SPU inserts and deletes flow entries in the flow table based on policy matching results. In the data plane, the IOC3 parses packets, and looks them up in the flow table. If the IOC3 finds a match in the flow table, then it forwards packets based on the instructions given in the flow table. The IOC3 can perform NAT, encapsulate the Level 2 (L2) header, and forward the packets out of the egress interface. The egress interface can be located on the same IOC3 (intra-card case) or on another IOC3 (inter-card case).

> **NOTE:** Flow table lookup in the IOC3 occurs only in ingress. Egress datapath packet handling is the same as supported in the previous release.

When the IOC3 receives the first packet, it does not match any existing fast-forward session. The default hash-based forwarding is performed to send the first packet to the SPU. The SPU then creates the security session. If the SPU finds that the traffic is qualified for fast forwarding, and the related IOC3 supports fast forwarding, it will install fast-forward session to the IOC3. If fast forwarding cannot be applied to the traffic, no session message is sent, and the IOC3 uses the default hash-based forwarding to forward the packets to the SPU.

In fast-forward IOC3 processing, if a fast-forward session is matched, the packet can be directly forwarded according to the session flow result. The IOC3 takes all the necessary actions, including forwarding the packet, TTL checking and decreasing, NAT translation, L2 header encapsulation and so on.

In addition, the XL chip sends one copy of the forwarding packet to the SPU at every predefined time. This copy is used to refresh the SPU session, detect the current XL chip state, and so on. The SPU consumes this packet and does not forward it, because the real packet has been processed and transmitted.

Expres Path support on IOC3 is illustrated in Figure 31 on page 262, Figure 32 on page 262, and Figure 33 on page 263.

Figure 31: IOC3 Intra-PFE Express Path

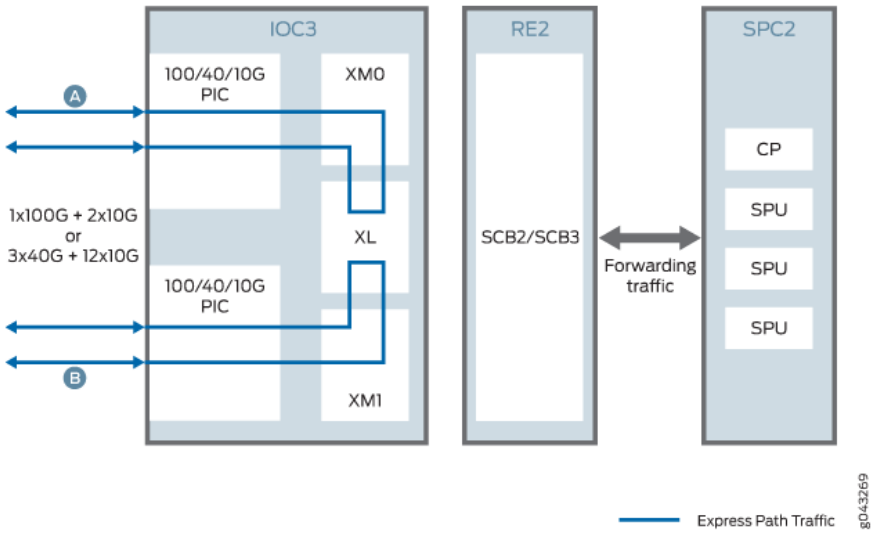


Figure 32: IOC3 Inter-PFE Express Path

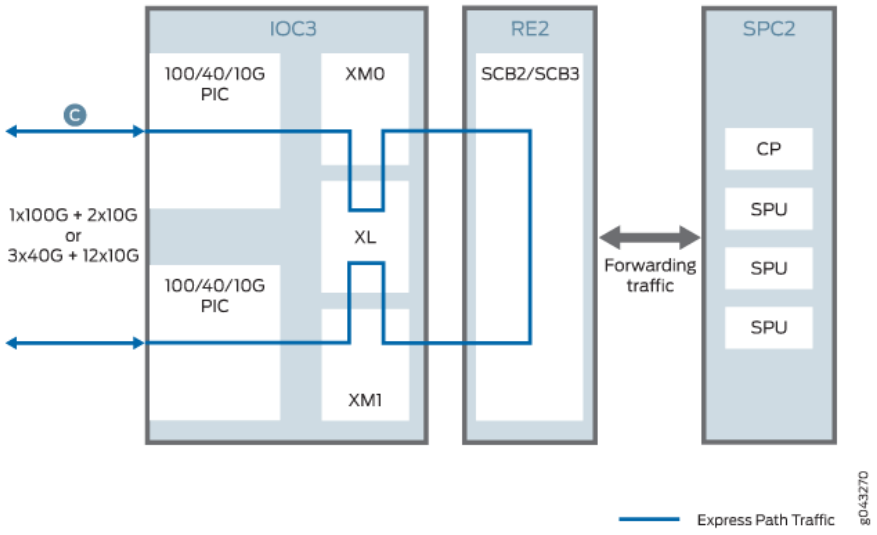
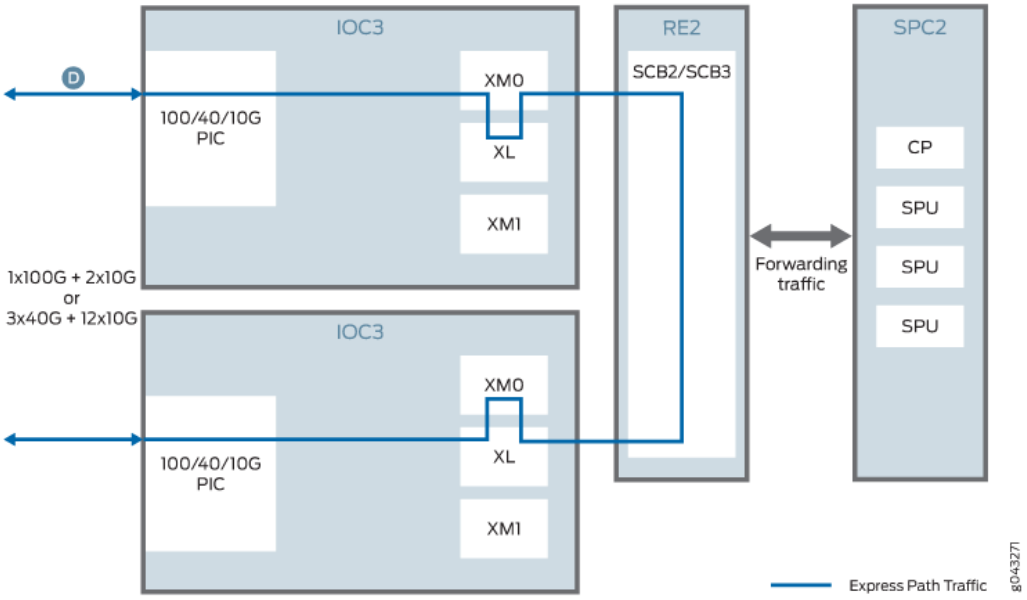


Figure 33: Inter-IOC3 Express Path



IPv6 Flow in Express Path Mode for IOC2 and IOC3

IPv6 traffic is supported on SRX5000 line devices with the SRX5K-MPC (IOC2), the SRX5K-MPC3-100G10G (IOC3), or the SRX5K-MPC3-40G10G (IOC3) in Express Path mode.

On SRX5000 line devices, Express Path for IPV6 traffic is not supported on legacy IOC cards. However, IPv6 regular flow mode is supported on legacy IOCs.

When an Express Path session is created on the network processor, subsequent packets of the flow match the session on the network processors. The network processor then processes and forwards the packet. The network processor also handles additional processing such as TCP sequence check, time-to-live (TTL) processing, and Layer 2 header translation.

The following features are not supported in Express Path mode:

- IPv6 NAT
- Transparent mode
- Configuring different MTU size values
- Class of Service (CoS) on egress interfaces

Note the following limitations:

- Express Path sessions for IPv6 traffic traversing on legacy IOC cards is not supported. IOC2 and IOC3 does not support IPv6 traffic in Express Path sessions when traffic traverse between legacy IOC and IOC2 or IOC3. Normal IPv6 traffic is still supported in this scenario.
- A redundant Ethernet interface must contain both child interfaces from the same IOC type. For example, if one child link is from 10-Gigabit Ethernet on IOC2, the second child link should also be from the IOC2. Similarly, both child interfaces can be from IOC3. Configuring child interfaces by mixing the links from both IOC2 and IOC3 is not supported.

IPv6 Flow in Express Path Mode

IPv6 traffic is supported on SRX5000 line devices with SRX5K-MPC (IOC2), the SRX5K-MPC3-100G10G (IOC3), or the SRX5K-MPC3-40G10G (IOC3). All IPv6 traffic is handled in regular flow mode, meaning that packets are forwarded to the SPU for flow processing. Egress IPv6 traffic is also forwarded from the SPU to the network processor, and then the network processor handles this traffic as regular flow traffic in the egress path.

When an Express Path session is created on the network processor, subsequent packets of the flow match the session on the network processors. The network processor then processes and forwards the packet. The network processor also handles additional processing such as TCP sequence check, time to live (TTL) processing, Network Address Translation (NAT), and Layer 2 header translation.

> **NOTE:** On the SRX5000 line devices, the Express Path sessions for IPv6 traffic that traverse between legacy IOC cards and not supported.
>
> The Express Path sessions for IPv6 traffic traversing only on legacy IOC cards or only on the SRX5K-MPC (IOC2), the SRX5K-MPC3-100G10G (IOC3), or the SRX5K-MPC3-40G10G (IOC3) are supported.

Understanding the Express Path Solution

The high-end SRX Series devices have long packet-processing latency because the packets are processed through the Services Processing Unit (SPU) and through several stages of buffers in the data path.

This feature introduces a local forwarding solution where the fast-path packets are processed by the network processor on the I/O Card (IOC), without going through the switch fabric or the SPU. This solution reduces the packet-processing latency.

The behavior of the network processor in different scenarios is as follows:

- **First-path flow**—The first-path flow is the same as the current network processor flow process. When the first packet arrives at the network processor, the network processor parses the TCP or the UDP packet to extract a 5-tuple key and then performs session lookup in the flow table. The network processor then forwards the first packet to the central point. The central point cannot find a match at this time, because this is the first packet. The central point and the SPU create a session and match it against user-configured policies to determine if the session is a normal session or a services-offload session.

  If the user has specified the session to be handled with services offload, the SPU creates a session entry in the network processor flow table, enabling the services-offload flag in the session entry table; otherwise, the SPU creates a normal session entry in the network processor without the services-offload flag.

- **Fast-path flow**—After the session entry is created in the network processor, subsequent packets of the session will match the session entry table.
  - If the services-offload flag is not set, then the network processor forwards the packet to the SPU specified in the session entry table. The packet goes through the normal flow process.
  - If the network processor finds the services-offload flag in the session entry table, it will process the packet locally and send the packet out directly.

    NOTE: The fast-forwarding function on the network processor supports one-fanout multicast sessions. The egress port in the session must also be associated with the same network processor of the ingress port. All other multicast cases need to be handled as normal sessions.

- **NAT process**—The SPU is responsible for mapping between the internal IP address or port and the external IP addres or port. When the first packet of the session arrives, the SPU allocates the IP address or port mapping and stores the information in the network processor session entry. The network processor does the actual packet modification if the NAT flag is set.

- **Session age-out**—To improve traffic throughput for services-offload sessions, a copy of a packet is sent to the SPU at every predefined time period to reduce the packet processing demand on the SPU. To limit the number of packet copies sent to the SPU, a timestamp is implemented for each services-offload session. This enables the network processor to calculate the elapsed time since the last session match.

If the elapsed time is greater then the predefined time period, then the network processor sends a copy of the packet to the SPU, and updates the session timestamp.

- **Session termination and deletion**—If the network processor receives an IP packet with a FIN (finished data) or a RST (reset connection) flag, it forwards the packet to the SPU. The SPU then deletes the session cache on network processor. The network processor continues to receive and forward any packets to the SPU during state transition.

Enabling and Disabling Express Path

Express Path (formerly known as services offloading) is a mechanism for processing fast-path packets in the network processor instead of in the Services Processing Unit (SPU). This method reduces the packet-processing latency that arises when packets are forwarded from network processors to SPUs for processing and back to I/O cards (IOCs) for transmission.

> **NOTE:** Starting with Junos OS Release 12.3X48-D10 and Junos OS Release 17.3R1, the Express Path license is no longer required to enable the Express Path functionality. Your previously acquired Express Path license will not be effective anymore. (Prior to Junos OS Release 12.3X48-D10, Express Path was a licensed software feature, formerly known as “services offloading.”)

- When device is operating in chassis cluster mode, you need to reboot both the nodes when changing FPC(s) to Express Path mode.
- During initialization, when a network processor is configured to perform Express Path, then the FPC CPU will load a special image to the network processor.

You can enable Express Path mode as follows:

- Set the Express Path mode on the selected card.
- Reboot the device containing the Express Path network processor to load the Express Path firmware image on the network processors.
- Configure a policy to define the traffic that should take fast-path.

To configure the Express Path mode:

- For configuring Express Path on an SRX5000 line device with a IOC1 or FIOC cards, use the **set chassis fpc *fpc-number* pic *pic-number* services-offload** command.
- For configuring Express Path on an SRX5000 line device with Modular Port Concentrator (MPC), enable NP cache on the IOC using the **set chassis fpc *fpc-number* np-cache** command.

> **NOTE:** The **set or delete chassis fpc *fpc-number* services-offload** command is deprecated.

- On SRX4600 device, the **np-cache** option is enabled by default. Hence, the **set chassis fpc *fpc-number* np-cache** command is not applicable.
- To disable Express Path on an SRX5000 line device with Modular Port Concentrator (MPC), use the **delete chassis fpc *fpc-number* np-cache** command.
- Reboot the device when Express Path is disabled.

SEE ALSO

Understanding the Express Path Solution | 265

Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path | 269

Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path | 272

Example: Enabling Express Path in Security Policies

IN THIS SECTION

- Requirements | 267
- Overview | 268
- Configuration | 268
- Verification | 269

This example shows how to enable Express Path (formerly known as *services offloading*) in security policies.

Requirements

Before you begin, understand the Express Path overview..

Overview

In this example, you enable Express Path in security policies to specify whether the traffic qualifies for Express Path.

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match source-address 192.0.2.2
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match destination-address
    198.51.100.10
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match application junos-http
set security policies from-zone untrust to-zone trust policy services-offload-pol1 then permit services-offload
```

Step-by-Step Procedure

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

To configure Express Path in policies:

1. Configure a policy to process the traffic that goes to the HTTP static ports.

```
[edit security policies from-zone untrust to-zone trust policy services-offload-pol1]
user@host# set match source-address 192.0.2.2
user@host# set match destination-address 198.51.100.10
user@host# set match application junos-http
```

2. Enable Express Path in the security policy.

```
[edit security policies from-zone untrust to-zone trust policy services-offload-pol1]
user@host# set then permit services-offload
```

Results

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

```
[edit]
user@host# show security policies
from-zone untrust to-zone trust {
    policy services-offload-pol1 {
        match {
            source-address 192.0.2.2;
            destination-address 198.51.100.10;
            application junos-http;
        }
        then {
            permit {
                services-offload;
            }
        }
    }
}
```

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

Verification

Verifying Express Path in Policies

Purpose

Verify that Express Path is enabled.

Action

From operational mode, enter the **show security policies** command.

Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path

IN THIS SECTION

- Requirements | **270**
- Overview | **270**
- Configuration | **270**
- Verification | **271**

This example shows how to configure an IOC on SRX5000 line of devices to support Express Path (formerly known as *services offloading*).

Requirements

Before you begin, understand the Express Path overview.

Overview

In this example, you configure the IOC on SRX5000 line devices to perform Express Path.

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, and then enter **commit** from configuration mode.

```
set chassis fpc 3 pic 0 services-offload
```

NOTE: For SRX5000 line devices, the IOC slot number is 3.

Step-by-Step Procedure

To configure the IOC you need to run the following commands:

1. Set the services offload mode on the IOC.

```
[edit]
user@host# set chassis fpc 3 pic 0 services-offload
```

2. Commit the configuration.

```
[edit]
user@host# commit
warning: System restart is required after fpc 3 pic 0 changed to services-offload
 mode.
commit complete
```

3. Reboot the device.

Results

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

```
[edit]
user@host> show chassis fpc pic-status
```

```
Slot 0   Online       SRX5k SPC
  PIC 0  Online       SPU Cp-Flow
  PIC 1  Online       SPU Flow
Slot 1   Online       SRX5k FIOC
  PIC 0  Online       4x 10GE XFP
  PIC 1  Online       16x 1GE SFP
Slot 3   Online       SRX5k DPC 4X 10GE
  PIC 0  Online       1x 10GE(LAN/WAN) RichQ- services-offload
  PIC 1  Online       1x 10GE(LAN/WAN) RichQ
  PIC 2  Online       1x 10GE(LAN/WAN) RichQ
  PIC 3  Online       1x 10GE(LAN/WAN) RichQ
Slot 5   Online       SRX5k DPC 40x 1GE
  PIC 0  Online       10x 1GE RichQ
  PIC 1  Online       10x 1GE RichQ
  PIC 2  Online       10x 1GE RichQ
  PIC 3  Online       10x 1GE RichQ
```

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

Verification

Verifying the Configuration of IOC for Express Path

Purpose

Verify that the IOC was configured properly for Express Path.

Action

From operational mode, enter the **show chassis fpc pic-status** command.

Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path

IN THIS SECTION

- Requirements | 272
- Overview | 272
- Configuration | 272
- Verification | 274

This example shows how to configure an SRX5K-MPC on an SRX5000 line device to support Express Path (formerly known as *services offloading*).

Requirements

This example uses the following hardware and software components:

- One SRX5000 line device with an SRX5K-MPC
- Junos OS Release 12.3X48 or later for SRX Series devices

Before you begin, understand Express Path overview..

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

Overview

In this example, you configure the SRX5K-MPC on an SRX5000 line device to perform NP cache and Express Path.

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, and then enter **commit** from configuration mode.

```
set chassis fpc 2 np-cache
set chassis fpc 3 np-cache
```

```
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match source-address
   198.51.100.20
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match destination-address
   192.0.2.11
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match application junos-http
set security policies from-zone untrust to-zone trust policy services-offload-pol1 then permit services-offload
```

Step-by-Step Procedure

To configure an SRX5K-MPC on an SRX5000 line device to perform Express Path:

1. Set NP cache mode on the SRX5K-MPC on FPC 1 and FPC 2.

```
[edit]
user@host# set chassis fpc 2 np-cache
user@host# set chassis fpc 3 np-cache
```

2. Configure a policy to process the traffic that goes to the HTTP static ports.

```
[edit security policies from-zone untrust to-zone trust policy services-offload-pol1]
user@host# set match source-address 198.51.100.20
user@host# set match destination-address 192.0.2.11
user@host# set match application junos-http
```

3. Enable Express Path in the security policy.

```
[edit security policies from-zone untrust to-zone trust policy services-offload-pol1]
user@host# set then permit services-offload
```

4. Commit the configuration.

```
[edit]
user@host# commit
```

```
warning: System or cluster nodes need to reboot after fpc 3 changed to np-cache
 mode.
```

5. Reboot the device.

Results

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

For brevity, this **show** command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
...
   fpc 2 {
      np-cache;
   }
   fpc 3 {
      np-cache;
   }
...
```

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

Verification

Confirm that the configuration is working properly.

Verifying the Configuration of an SRX5K-MPC for Express Path

Purpose

Verify that the SRX5K-MPC was configured properly for Express Path.

Action

From operational mode, enter the **show chassis fpc pic-status** command.

```
Slot 0   Online       SRX5k DPC 40x 1GE
  PIC 0  Online       10x 1GE RichQ
  PIC 1  Online       10x 1GE RichQ
  PIC 2  Online       10x 1GE RichQ
  PIC 3  Online       10x 1GE RichQ
Slot 2   Online       SRX5k IOC II
  PIC 0  Online       2x 40GE QSFP+- np-cache/services-offload
Slot 3   Online       SRX5k IOC II
  PIC 0  Online       10x 10GE SFP+- np-cache/services-offload
  PIC 2  Online       10x 10GE SFP+- np-cache/services-offload
Slot 5   Online       SRX5k SPC
  PIC 0  Online       SPU Cp-Flow
  PIC 1  Online       SPU Flow
```

Meaning

The output provides the status of PICs with Express Path enabled on them.

Example: Configuring SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) on an SRX5000 Line Device to Support Express Path

IN THIS SECTION

- Requirements | 275
- Overview | 275
- Configuration | 275
- Verification | 278

This example shows how to configure an SRX5K-MPC3-100G10G (IOC3) or an SRX5K-MPC3-40G10G (IOC3) on an SRX5000 line device to support Express Path (formerly known as *services offloading*).

Requirements

This example uses the following hardware and software components:

- One SRX5000 line device with an SRX5K-MPC3-40G10G (IOC3)
- Junos OS Release 15.1X49-D10 or later for SRX Series devices

Before you begin, understand Express Path overview..

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

Overview

In this example, you configure an SRX5K-MPC3-40G10G (IOC3) on an SRX5000 line device to perform Express Path.

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set chassis fpc 4 np-cache
set chassis fpc 5 np-cache
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match source-address 192.0.2.41
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match destination-address
    203.0.113.10
set security policies from-zone untrust to-zone trust policy services-offload-pol1 match application junos-http
set security policies from-zone untrust to-zone trust policy services-offload-pol1 then permit services-offload
```

Step-by-Step Procedure

To configure an SRX5K-MPC3-40G10G (IOC3) on an SRX5000 line device to perform Express Path:

1. Set the Express Path mode on the SRX5K-MPC3 on FPC 4 and FPC 5.

```
[edit]
user@host# set chassis fpc 4 np-cache
user@host# set chassis fpc 5 np-cache
```

2. Configure a policy to process the traffic that goes to the HTTP static ports.

```
[edit security policies from-zone untrust to-zone trust policy services-offload-pol1]
user@host# set match source-address 192.0.2.41
user@host# set match destination-address 203.0.113.10
user@host# set match application junos-http
```

3. Enable Express Path in the security policy.

```
[edit security policies from-zone untrust to-zone trust policy services-offload-pol1]
user@host# set then permit services-offload
```

4. Commit the configuration.

```
[edit]
user@host# commit
```

```
warning: System or cluster nodes need to reboot after fpc 3 changed to np-cache
 mode.
```

5. Reboot the device.

Results

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

For brevity, this **show** command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
...
   fpc 4{
      services-offload;
   }
   fpc 5{
      services-offload;
   }
...
...
   from-zone <fzone> to-zone <tzone> {
   policy <policy-name> {
      match {
         <match-tuples>
      }
      then {
         action (
         permit {
            ...
            services-offload
            ^^^^^^^^^^^^^^^^
         }
         reject
         deny
         log
         );
      }
      scheduler-name <scheduler-name>;
   }
   ...
}
```

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

Verification

Confirm that the configuration is working properly.

Verifying the Configuration of an SRX5K-MPC3 (IOC3) for Express Path

Purpose

Verify that the SRX5K-MPC3-40G10G (IOC3) was configured properly for Express Path.

Action

From operational mode, enter the **show chassis fpc pic-status** command.

```
Slot 0   Offline      SRX5k DPC 40x 1GE
Slot 1   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 2   Offline      SRX5k SPC
Slot 4   Online       SRX5k IOC3 24XGE+6XLG
  PIC 2  Online       3x 40GE QSFP+- np-cache/services-offload
  PIC 3  Online       3x 40GE QSFP+- np-cache/services-offload
Slot 5   Online       SRX5k IOC II
  PIC 0  Online       10x 1GE(LAN) SFP- np-cache/services-offload
  PIC 1  Online       10x 1GE(LAN) SFP- np-cache/services-offload
  PIC 2  Online       10x 10GE SFP+- np-cache/services-offload
```

Meaning

The output provides the status of PICs with Express Path enabled on them.

Example: Configuring Express Path on an SRX5000 Line Device with IOC3

IN THIS SECTION

- Requirements | **279**
- Overview | **279**

Configuration | 279
Verification | 283

This example shows how to configure Express Path (formerly known as *services offloading*) on an SRX5K-MPC3-100G10G (IOC3) or an SRX5K-MPC3-40G10G (IOC3) on an SRX5000 line device.

Express Path is a mechanism for processing fast-path packets in the network instead of in the Services Processing Unit (SPU). This method reduces the long packet-processing latency that arises when packets are forwarded from network processors to SPUs for processing and back to IOCs for transmission.

Starting in Junos OS Release 15.1X49-D40, the configuration is valid for IPv6 traffic, whereas before it was supported for IPv4 traffic only.

Requirements

This example uses the following hardware and software components:

- One SRX5000 line device with an IOC3 card
- Junos OS Release 15.1X49-D40 or later for SRX Series devices

Before you begin, understand the Express Path overview..

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

Overview

In this example, you configure Express Path on IOC3 on an SRX5000 line device for IPv6 traffic.

You configure two interfaces on IOC3 card and assign IPv6 addresses to them. Then you enable flow-based processing for IPv6 traffic. Next, you set up zones and add interfaces to them. Then you provide communication between the two different zones by configuring a security policy to allow traffic between two zones. You also enable Express Path in security policies to specify whether the traffic qualifies for Express Path.

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from configuration mode.

```
set interfaces et-2/1/0 unit 0 family inet6 address 2001:db8::4:12/32
set interfaces et-2/3/0 unit 0 family inet6 address 2001:db8::6:11/32
set security forwarding-options family inet mode flow-based
set security forwarding-options family inet6 mode flow-based
set zones security-zone zone-1 host-inbound-traffic system-services all
set zones security-zone zone-1 host-inbound-traffic protocols all
set zones security-zone zone-1 interfaces et-2/1/0.0
set zones security-zone zone-2 host-inbound-traffic system-services all
set zones security-zone zone-2 host-inbound-traffic protocols all
set zones security-zone zone-2 interfaces et-2/3/0.0
security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 match source-address any
security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 match destination-address any
security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 match application any
security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 then permit then permit
    services-offload
security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 match source-address any
security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 match destination-address any
security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 match application any
security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 then permit then permit
    services-offload
set chassis fpc 2 np-cache
```

Step-by-Step Procedure

To configure an SRX5K-MPC3-40G10G (IOC3) on an SRX5000 line device to perform Express Path:

1. Configure Ethernet interface and assign an IPv6 address to it.

```
[edit]
user@host# set interfaces et-2/1/0 unit 0 family inet6 address 2001:db8::4:12/32
user@host# set interfaces et-2/3/0 unit 0 family inet6 address 2001:db8::6:11/32
```

2. Enable flow-based processing for IPv6 traffic.

```
[edit]
user@host# set security forwarding-options family inet mode flow-based
user@host# set security forwarding-options family inet6 mode flow-based
```

3. Configure security zones and add interfaces and allow all system services and interfaces. Configure a security zone and specify the types of traffic and protocols that are allowed on interface et-2/1/0.0.

```
[edit]
```

```
user@host# set zones security-zone zone-1 host-inbound-traffic system-services all
user@host# set zones security-zone zone-1 host-inbound-traffic protocols all
user@host# set zones security-zone zone-1 interfaces et-2/1/0.0
```

4. Configure security zones and add interfaces and allow all system services and interfaces. Configure a security zone and specify the types of traffic and protocols that are allowed on interface et-2/3/0.0.

```
[edit]
user@host# set zones security-zone zone-2 host-inbound-traffic system-services all
user@host# set zones security-zone zone-2 host-inbound-traffic protocols all
user@host# set zones security-zone zone-2 interfaces et-2/3/0.0
```

5. Create a policy and specify the match criteria for that policy. The match criteria specifies that the device can allow traffic from any source, to any destination, and on any application. Enable Express Path in the security policy.

> **NOTE:** You can specify the wildcard any-ipv6 for the source and destination address match criteria to include only IPv6 addresses. Specifying any option for the source and destination address match criteria to include both IPv4 and IPv6 addresses.

```
[edit security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2]
user@host# set security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 match
    source-address any
user@host# set security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 match
    destination-address any
user@host# security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 match application
    any
user@host# security policies from-zone zone-2 to-zone zone-1 policy express-path-policy-2 then permit
    services-offload
```

```
[edit]
user@host# set security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 match
    source-address any
user@host# set security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 match
    destination-address any
user@host# set security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 match
    application any
```

```
user@host# set security policies from-zone zone-1 to-zone zone-2 policy express-path-policy-1 then permit
  services-offload
```

6. Set the Express Path mode on IOC3.

```
[edit]
user@host# set chassis fpc 2 np-cache
```

Results

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

```
from-zone zone-1 to-zone zone-2 {
   policy express-path-policy--1 {
      match {
         source-address any;
         destination-address any;
         application any;
      }
      then {
         permit {
            services-offload;
         }
      }
   }
}
from-zone express-path-policy--2 {
   policy policy-2 {
      match {
         source-address any;
         destination-address any;
         application any;
      }
      then {
         permit {
            services-offload;
         }
      }
   }
}
```

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

Verification

Confirm that the configuration is working properly.

Verifying the Configuration of an SRX5K-MPC3 (IOC3) for Express Path

Purpose

Verify that the IOC3 was configured properly for Express Path.

Action

From operational mode, enter the **show chassis fpc pic-status** command.

```
Slot 1   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 2   Online       SRX5k IOC3 2CGE+4XGE
  PIC 0  Online       2x 10GE SFP+- np-cache/services-offload
  PIC 1  Online       1x 100GE CFP2- np-cache/services-offload
  PIC 2  Online       2x 10GE SFP+- np-cache/services-offload
  PIC 3  Online       1x 100GE CFP2- np-cache/services-offload
Slot 3   Online       SRX5k IOC3 24XGE+6XLG
  PIC 0  Offline      12x 10GE SFP+
  PIC 1  Offline      12x 10GE SFP+
  PIC 2  Online       3x 40GE QSFP+- np-cache/services-offload
  PIC 3  Online       3x 40GE QSFP+- np-cache/services-offload
Slot 4   Offline      SRX5k IOC3 24XGE+6XLG
```

Meaning

The output provides the status of PICs with Express Path enabled on them.

Verifying All Active Sessions on the Device

Purpose

Display information about all currently active Express Path security sessions on the device.

Action

From operational mode, enter the **show security flow session services-offload** command.

```
Flow Sessions on FPC1 PIC1:

Session ID: 50000002, Policy name: express-path-policy-2/5, Timeout: 60, Valid
  In: 2001:db8::4:12/32 --> 2001:db8::6:11/32;udp, If: et-2/3/0.0, Conn ID: 0x0,
Pkts: 181
29505, Bytes: 1740432530, CP Session ID: 50000002
  Out: 2001:db8::6:11/32 --> 2001:db8::4:12/32;udp, If: et-2/1/0.0, Conn ID: 0x0,
 Pkts: 18
129505, Bytes: 1740432530, CP Session ID: 50000002
Total sessions: 1
```

Meaning

The output provides the policy details for sessions on which Express Path was enabled.

Example: Configuring Low Latency

IN THIS SECTION

- Requirements | **285**
- Overview | **285**
- Configuration | **285**
- Verification | **286**

The low latency feature allows you to configure the mode of the network processor's traffic manager (TM) on the egress path. If low latency is enabled, the network processor is initialized without the traffic manager, thus reducing the overall latency in the Express Path (formerly known as *services offloading*).

> **NOTE:** Because all SRX Series CoS functions are supported by the traffic manager, CoS functions are not supported when low latency is enabled.

Low latency reduces the total NPC integrated with an existing IOC (NP-IOC) latency by 0.7 us. This latency reduction brings the NP-IOC card total latency to 8.7 us. The low-latency feature is supported for intra-NP-IOC card traffic only; it is not applicable to inter-NP traffic.

In the low-latency mode, the network processor does not have an egress buffer at the traffic manager. Packets are delivered directly to the system packet interface (SPI) for the field-programmable gate array (FPGA) to process.

> **NOTE:** The low latency feature is only applicable to the NP-IOC card.

Requirements

Before you begin, understand Express Path overview..

This example uses the following software and hardware components:

- Junos OS Release 12.1X44-D10
- One SRX Series device
- One Services Processing Card (SPC)

Overview

In this example, you configure the network processor for low latency mode.

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, and then enter **commit** from configuration mode.

```
set chassis fpc 7 pic 0 services-offload low-latency
```

Step-by-Step Procedure

To enable low-latency mode:

1. Enable the Express Path mode on the NP-IOC.

```
[edit]
user@host# set chassis fpc 7 pic 0 services-offload low-latency
```

2. Commit the configuration.

```
[edit]
user@host# commit
```

```
warning: System restart is required after fpc 7 pic 0 changed to services-offload
 mode.
commit complete
```

3. Reboot the device.

Results

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

```
[edit]
user@host> show configuration chassis
```

```
fpc 7 {
     pic 0 {
         services-offload {
             low-latency;
         }
     }
}
```

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

Verification

Verifying Low Latency Configuration

Purpose

Verify that low-latency was enabled.

Action

From operational mode, enter the **show chassis fpc pic-status** command.

```
root@kg04> show chassis fpc pic-status
  Slot 0   Online       SRX5k SFB 12GE
    PIC 0  Online       8x 1GE-TX 4x 1GE-SFP
```

```
Slot 1   Online       SRX5k 2x10GE XFP
  PIC 0  Online       2x 10GE-XFP
Slot 2   Online       SRX5k SPC
  PIC 0  Online       SPU Cp-Flow
Slot 7   Online       SRX5k 2x10GE NP-IOC
  PIC 0  Online       2x 10GE-SFP+- services-offload low-latency
```

Managing Packet Fragmentation in IPsec VPN Networks

IN THIS SECTION

- Fragmentation Counters Feature Overview | **288**
- Understanding Fragmentation and MTU and MSS Sizes | **288**
- Using Fragmentation Counter Statistics to Tune Your System | **289**

Packet fragmentation degrades system performance in IPsec VPN networks. The SRX Series packet fragmentation counters feature allows you to monitor the amount of packet fragmentation incurred in processing traffic for IPsec tunnels on your device and throughout your network. It counts fragmented packets that can occur before tunnel encapsulation and afterward for individual tunnels. It also counts overall the number of fragmented packets for tunnel sessions on a Services Processing Unit (SPU).

To understand the amount of packet fragmentation in your network in order to prevent it from occurring, it is helpful to be able to measure it. After you tune your system for improvement, it is useful to be able to verify the results.

The fragmentation counters feature provides output through the **show security flow** commands that you can use to display fragmentation counter statistics. You can display fragmented packet numbers collectively for individual IPsec tunnels. You can also obtain a summary of the number of fragmented packets based on the SPU.

You can use the fragmentation information provided through **show** commands as input to your iterative tuning process to decrease the likelihood of fragmentation occurrence. Use of this feature allows you to achieve optimum SRX Series performance otherwise limited by packet fragmentation.

Fragmentation Counters Feature Overview

Datagrams are fragmented when a packet is larger than the maximum transmission unit (MTU) size established for a device's egress interface. The egress interface's MTU size determines the size of the packets sent to the receiving device. A datagram could also be fragmented into smaller packets to transit a link in the datapath because the packet is larger than the amount of data that the receiving device can accept or larger than the MTU of any link in the datapath. In any case, the packet header of the original datagram that was broken into fragments is added to each of the fragmented packets, in addition to the parts of the payload that the fragment carries.

It is important to understand the degree and kinds of fragmentation occurring on your device in order to tune your system to avoid it. The Junos OS for SRX Series fragmentation counters feature counts packet fragments for IPsec tunnels that can occur before and after a packet is encapsulated with an IPsec encryption header.

The fragmentation counters feature takes into account the following kinds of packet fragmentation:

Pre-fragmentation—Self-generated packet fragmentation that occurs prior to encapsulation

Post-fragmentation—Packets that are received by the SRX Series device and packets that are fragmented after encryption.

For an individual tunnel, a counter is increased whenever a fragment is encountered. Fragments that occur before packet encapsulation are counted separately from fragments that occur because of encapsulation. When a counter is increased for an individual tunnel, the SPU fragmentation counter is also increased.

Understanding Fragmentation and MTU and MSS Sizes

Packet fragmentation can negatively impact performance of the entire IPsec VPN, and it must be avoided for that reason. Fragmentation is likely to occur when a datagram approximates the MTU size set for the egress interface of the sending device. When IPsec VPN datagrams are fragmented, the resulting fragment packets are encapsulated with IPsec ESP or AH headers in addition to the datagram's original TCP header. Fragmentation negatively impacts the IPsec peers at either end of the IPsec VPN tunnel. Fragmentation—breaking a datagram into smaller packets to be reassembled later—incurs CPU and memory overhead on both the sending peer and the receiving peer. The impact on the sending peer is minimal. It must break down the datagram. The impact on the receiving peer is far greater because it must allocate memory for incoming packets and reassemble them into the complete datagram before it can decrypt the cohesive datagram.

The size of a packet to be transmitted to the receiving peer is based on two values: the MTU size and maximum segment size (MSS). The MTU size established for the egress interface determines the size of the datagram that the sending peer transmits to the receiving peer. Although a larger MTU size can result in greater efficiency, it can have a negative impact, resulting in packet fragmentation downstream.

The MSS of a device specifies the maximum amount of information that the device can accept in a single IP datagram. In IPsec VPNs, each peer compares its outgoing interface MTU size with its own MSS buffer size. It must send the smaller of the two values to the receiving peer as its MSS. During the three-way handshake negotiation between the two IPsec VPN peers, the smaller MSS value is selected to be used in sending packets. The MSS value is sent as a TCP header option in TCP SYN segments.

Using Fragmentation Counter Statistics to Tune Your System

There are a number of methods that you can use to limit the degree of fragmentation that can occur when IPsec VPN tunnels are used. Regardless of the method that you use, it is helpful to be able to observe and measure the volume of fragmentation that is being transmitted before and after you iteratively tune your network. The SRX Series fragmentation counters feature provides that information in the following **show** commands output:

- To see fragmentation information for individual tunnels, use the **show security flow session tunnel extensive** command.
- To see overall fragmentation information based on an SPU, use the **show security flow session tunnel summary** command.
- To see statistics on the number of pre-fragments and post-fragments on an SPU, use the **show security flow statistics** command.

Here are two of the basic approaches that you can take to manage fragmentation between the two IPsec VPN peers:

1. Manipulate the MSS of the sending peer to establish the appropriate MTU size for its egress interface.

   > **NOTE:** The sending peer device should not send packets that are larger than the receiving peer device can accept, as determined by the receiving peer's MSS value.

   For details on changing the MSS setting on the sending peer's device to effect a smaller MTU size on that device.

   Use the fragmentation counters statistics displayed by the related **show** commands to iteratively tune the MSS value on the sending peer until fragmentation between the peers is eliminated.

   To get the fragmentation statistics result of your tuning, you must renegotiate the MSS value with the receiving peer.

   Before you renegotiate with the receiving peer, you must clear the **show** commands to reset their counters. If further tuning is required, you must clear the **show** commands before changing the MSS value and renegotiating again with the receiving peer.

   To clear the fragmentation counters for the **show security flow statistics** command, use the following command:

```
user@host# clear security flow statistics
```

To clear the fragmentation counters for the **show security flow session tunnel extensive** and the **show security flow session tunnel summary** commands, you must deactivate the IPsec tunnel and then reactivate it.

Use the following statements to deactivate the IPsec VPN:

```
user@host# deactivate security ipsec
user@host# commit
```

Use the following statements to reactivate the IPsec VPN to enact the three-way handshake with the peer device in which the MSS values of the peers are exchanged:

```
user@host# activate security ipsec
user@host# commit
```

2. Use the ping command

   You could use the ICMP ping command to determine the correct packet size to use in establishing the appropriate MTU size. To find the proper MTU size, you must send the ping repeatedly to the receiving peer until no fragmentation message is returned.

   You could start at 1450 and if you receive a fragmentation message, you could decrease the size by 10 each time you issue the ping command. If you do not get a fragmented packet reply message, you could incrementally increase the MTU size.

   Although you can control fragmentation between the two IPsec VPN endpoint peers, it can happen that a link in the datapath between them cannot accept a packet because its MSS value is too small or a link could have a smaller MTU size than the size of the packet that it received and must break it down the before transmitting it. Technologies are available such as path maximum transmission unit (PMTU) that can be used to dynamically determine the MTU size to avoid fragmentation along the datapath.

SEE ALSO

show security flow session tunnel | **714**

show security flow statistics | **725**

CHAPTER 6

Configuration Statements

aging | 294

all-tcp | 295

allow-dns-reply | 296

allow-embedded-icmp | 297

allow-reverse-ecmp | 298

application-services (Security Forwarding Process) | 299

apply-to-half-close-state | 301

destination-header | 302

destination-port (Security Forwarding Options) | 304

destination-prefix (Security Forwarding Options) | 308

early-ageout | 309

error | 310

fin-invalidate-session | 314

flow (Security Flow) | 316

force-ip-reassembly | 322

forwarding-process | 323

fpc error | 325

fru-poweron-sequence | 327

gre-in | 328

gre-out | 329

high-watermark | 330

hop-by-hop-header | 331

icmpv6-malformed | 332

idle-timeout (System Services) | 333

inline-tap | 334

interface-in (Security Forwarding Options) | 335

interface-out (Security Forwarding Options) | 336

ipv4-template (Services) | 337

ipv6-extension-header | 338

ipv6-extension-header-limit | 340

ipv6-malformed-header | 341

ipv6-template (Services) | 342

low-latency | 343

low-watermark | 344

maximize-idp-sessions | 345

mirror-filter (Security Forwarding Options) | 346

mode (Security Forwarding Options) | 348

no-sequence-check | 350

np-cache (Flexible PIC Concentrator) | 351

output (Security Forwarding Options) | 352

packet-filter | 353

packet-log (Security Flow) | 355

packet-ordering-mode (Application Services) | 356

pending-sess-queue-length | 357

per-session-statistics | 358

pre-id-default-policy | 359

preserve-incoming-fragment-size | 361

propagate-settings | 362

protocol (Security Forwarding Options) | 363

resource-manager | 365

reverse-route-packet-mode-vr | 366

route-change-timeout | 367

rst-invalidate-session | 368

rst-sequence-check | 369

sampling | 370

services-offload | 371

session (System Services) | 372

session-limit (System Services) | 373

source-port (Security Forwarding Options) | 374

source-prefix (Security Forwarding Options) | 378

syn-flood-protection-mode | 379

tcp-initial-timeout | 380

tcp-mss (Security Flow) | 381

tcp-session | 383

time-wait-state | 385

traceoptions (Security) | 386

traceoptions (Security Flow) | 388

transport (Security Log) | 391

weight (Security) | 393

aging

Syntax

```
aging {
    early-ageout seconds;
    high-watermark percent;
    low-watermark percent;
}
```

Hierarchy Level

```
[edit security flow]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

The session aging mechanism accelerates the session time-out process when the percentage of entries in the session table exceeds the high-watermark setting, and then stops when the percentage of entries in the session table falls below the low-watermark setting.

Options

The remaining statements are explained separately. See the CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

all-tcp

Syntax

```
all-tcp mss value;
```

Hierarchy Level

```
[edit security flow tcp-mss]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Set the TCP maximum segment size (MSS) value to enable MSS override for all TCP packets in network traffic.

Options

mss *value*—TCP MSS value.

Range: 64 through 65,535 bytes

Required Privilege Level

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

RELATED DOCUMENTATION

tcp-mss (Security Flow) | 381

Understanding Traffic Processing on Security Devices | 25

allow-dns-reply

Syntax

```
allow-dns-reply;
```

Hierarchy Level

```
[edit security flow]
```

Release Information
Statement introduced in Junos OS Release 8.5.

Description
Allow an incoming Domain Name Service (DNS) reply packet without a matched request. By default, if an incoming UDP first-packet has dst-port 53, the device checks the DNS message packet header to verify that the query bit (QR) is 0, which denotes a query message. If the QR bit is 1, which denotes a response message, the device drops the packet, does not create a session, and increments the illegal packet flow counter for the interface. Using the **allow-dns-reply** directs the device to skip the check.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

allow-embedded-icmp

Syntax

```
allow-embedded-icmp;
```

Hierarchy Level

```
[edit security flow]
```

Release Information
Statement introduced in Junos OS Release 12.3X48-D10.

Description
Allow ICMP error packets to pass through the device even when there is no session match for the embedded packet. Once enabled, all packets encapsulated in ICMP pass through and no policy affects this behavior. This feature is useful when you have asymmetric routing in your network and you want to use traceroute and other ICMP applications on your device.

The default behavior is to inspect the IP packet which is embedded in the ICMP error packet. If the embedded packet is part of an existing session, the ICMP packet is allowed to pass through. If there is no match, it is dropped. Without the **allow-embedded-icmp** option configured, the default behavior applies.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

allow-reverse-ecmp

Syntax

```
allow-reverse-ecmp
```

Hierarchy Level

```
[edit security flow]
```

Release Information
Statement introduced in Junos OS Release 17.3.

Description
Enable ECMP support for reverse traffic. In this case, Junos OS for SRX Series devices and vSRX instances use a hash algorithm to determine the interface to use for reverse traffic in a flow. This process is similar to asymmetric routing in which a packet traverses from a source to a destination in one path and takes a different path when it returns to the source.

If you do not enable this feature, the software selects a route in the ECMP set to the incoming interface for reverse traffic, which is the default behavior.

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

RELATED DOCUMENTATION

Understanding ECMP Flow-Based Forwarding | 86

Understanding ECMP Flow-Based Forwarding for Reverse Traffic on SRX Series Devices and vSRX Instances

application-services (Security Forwarding Process)

Syntax

```
application-services {
    enable-gtpu-distribution;
    maximize-alg-sessions;
    maximize-idp-sessions {
        weight (firewall | idp);
    }
    packet-ordering-mode (Application Services) {
        (hardware | software);
    }
}
```

Hierarchy Level

```
[edit security forwarding-process]
```

Release Information

Statement introduced in Junos OS Release 9.6. Statement updated in Junos OS Release 10.4. Statement updated in Junos OS Release 15.1X49-D40 with the **enable-gtpu-distribution** option.

Description

You can configure SRX4100, SRX4200, SRX5400, SRX5600, and SRX5800 devices to switch from an integrated firewall mode to maximize Intrusion Detection and Prevention (IDP) mode to run IDP processing in tap mode and increase the capacity of processing with the **maximize-idp-sessions** option. Inline tap mode can only be configured if the forwarding process mode is set to **maximize-idp-sessions**, which ensures stability and resiliency for firewall services. You also do not need a separate tap or span port to use inline tap mode. When you maximize IDP, you are decoupling IDP processes from firewall processes, allowing the device to support the same number of firewall and IDP sessions, also run the IDP processing in tap mode.

You can configure maximum Application Layer Gateway (ALG) sessions by using the **maximize-alg-sessions** option. The session capacity number for Real-Time Streaming Protocol (RTSP), FTP, and Trivial File Transfer Protocol (TFTP) ALG varies per flow SPU. For SRX5000 series devices the session capacity is 10,240 per flow SPU. You must reboot the device (and its peer in chassis cluster mode) for the configuration to take effect. The **maximize-alg-sessions** option now enables you to increase defaults as follows:

- TCP proxy connection capacity: 40,000 per flow SPU

  Flow session capacity is reduced to half per flow SPU; therefore the aforementioned capacity numbers will not change on central point flow.

Enable GPRS tunneling protocol. GTP-U session distribution is a UE (User equipment) based distribution, generating tunnel based GTP-U session and distributing them across SPUs on a UE basis.

Before 15.1X49-D40, GTP-U sessions are distributed by GGSN IP address always.

15.1X49-D40 onward, the GTP-U distribution is disabled and fat GTP-U sessions are distributed as normal UDP.

Use the **enable-gtpu-distribution** command to enable GTP-U session distribution.

Options

The remaining statements are explained separately. See the CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

apply-to-half-close-state

Syntax

```
apply-to-half-close-state;
```

Hierarchy Level

```
[edit security flow tcp-session time-wait-state]
```

Release Information
Statement introduced in Junos OS Release 12.1X46-D10.

Description
Configure the TCP session timeout in a half-closed state. This enables the system to apply the configured session timeout on receiving only one FIN packet (either client-to-server or server-to-client). When this is not configured, the default behavior takes effect—applying the configured TCP session timeout on receiving both the FIN packets. The default session timeout remains 150 seconds.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

destination-header

Syntax

```
destination-header {
    ILNP-nonce-option;
    home-address-option;
    line-identification-option;
    tunnel-encapsulation-limit-option;
    user-defined-option-type low | <to high>;
}
```

Hierarchy Level

```
[edit security screen ids-option screen-name ip ipv6-extension-header]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Define the IPv6 destination header screen option.

Options

ILNP-nonce-option—Enable the Identifier-Locator Network Protocol nonce screen option.

home-address-option—Enable the home address screen option.

line-identification-option—Enable the line identification screen option.

tunnel-encapsulation-limit-option—Enable the tunnel encapsulation limit screen option.

user-defined-header-type *low* | <to *high*>—Define the type of header range.

Range: 1 through 255.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding IPv6 Support for Screens

ipv6-extension-header | 338

hop-by-hop-header | 331

destination-port (Security Forwarding Options)

Syntax

```
destination-port port-number;
```

Hierarchy Level

```
[edit security forwarding-options mirror-filter filter-name]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Specify a Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) destination port number to be matched for mirroring. You can specify a numeric value or one of the text synonyms listed in Table 19 on page 305.

Table 19: Ports Supported by Services Interfaces

Port Name	Corresponding Port Number
afs	1483
bgp	179
biff	512
bootpc	68
bootps	67
cmd	514
cvspserver	2401
dhcp	67
domain	53
eklogin	2105
ekshell	2106
excc	512
finger	79
ftp	21
ftp-data	20
http	80
https	443
ident	113
imap	143
kerberos-sec	88
klogin	543

Table 19: Ports Supported by Services Interfaces *(continued)*

Port Name	Corresponding Port Number
kpasswd	761
krb-prop	754
krbupdate	760
kshell	544
ldap	389
ldp	646
login	513
mobileip-agent	434
mobilip-mn	435
msdp	639
netbios-dgm	138
netbios-ns	137
netbios-ssn	139
nfsd	2049
nntp	119
ntalk	518
ntp	123
pop3	110
pptp	1723
printer	515
radacct	1813

Table 19: Ports Supported by Services Interfaces *(continued)*

Port Name	Corresponding Port Number
radius	1812
rip	520
rkinit	2108
smtp	25
snmp	161
snmp-trap	162
snpp	444
socks	1080
ssh	22
sunrpc	111
syslog	514
tacacs	49
tacacs-ds	65
talk	517
telnet	23
tftp	69
timed	525
who	513
xdmcp	177

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346
show security forwarding-options mirror-filter | 736

destination-prefix (Security Forwarding Options)

Syntax

```
destination-prefix destination-prefix;
```

Hierarchy Level

[edit security forwarding-options mirror-filter *filter-name*]

Release Information
Statement introduced in Junos OS Release 12.1X46-D10.

Description
Specify the destination IP prefix or address to be matched for mirroring.

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

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346
show security forwarding-options mirror-filter | 736

early-ageout

Syntax

```
early-ageout seconds;
```

Hierarchy Level

```
[edit security flow aging]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Define the value before the device aggressively ages out a session from its session table.

Options

seconds —Amount of time that elapses before the device aggressively ages out a session.

Range: 1 through 65,535 seconds

Default: 20 seconds

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

error

Syntax

```
error {
    (fatal | major | minor) {
    threshold threshold number;
    action (alarm | disable-pfe | offline-pic | log | get-state |offline | reset);
    }
}
```

Hierarchy Level

[edit chassis]

[edit chassis fpc slot-number]

Release Information

Statement introduced in Junos OS Release 15.1X49-D40.

Description

Configure the threshold at which FPC errors will take the action you configure to be performed by the device.

Some devices include an internal framework for detecting and correcting FPC errors that can have the potential to affect services. You can classify FPC errors according to severity, set an automatic recovery action for each severity, and set a threshold (the number of times the error must occur before the action is triggered).

However, the alarm is added to the default fault handling action list for a fatal error. Adding an alarm to the default fault handling list will allow the chassis alarm to trigger the RG1 switchover as soon as the fatal error is detected.

Starting with Junos OS Release 15.1-X49-D50 and Junos OS Release 17.3R1, this feature supports I/O cards (IOCs) and Services Processing Cards (SPCs) on the Junos OS SRX5000 line of devices. The following cards are supported:

- IOC2
- IOC3
- SPC2

CAUTION: Take care when setting the fault handling actions for SPC2 cards on the SRX5000 line of devices. Consider that if you set the fault handling action on an SPC2 card to offline or reset, when the card is either taken offline or the reboot occurs, the chassis daemon (chassisd) will reboot all of its FPC cards, both SPCs and IOCs—that is, the entire chassis will be rebooted.

Options

You can configure the threshold for the following severity levels:

> **NOTE:** You cannot change the severity level of an error.

- **fatal**—Fatal error on the FPC. An error that results in the blockage of a considerable amount of traffic across modules is a fatal error. (default: raise an alarm and reset the FPC)
- **major**—Major error on the FPC. An error that results in continuing loss of packet traffic but does not affect other modules is a major error. (default: get the current state of the FPC and raise an alarm)
- **minor**—Minor error on the FPC. An error that results in the loss of a single packet but is fully recoverable is a minor error. (default: write a log for the event.)
- **threshold *threshold-value***—Configure the threshold value at which to take action. If the severity level of the error is fatal, the action is carried out only once when the total number of errors exceeds the threshold value. If the severity level of the error is major, the action is carried out once after the occurrence exceeds the threshold. If the severity level is minor, the action is carried out as many times as the value specified by the threshold. For example, when the severity level is minor, and you have configured the threshold value as 10, the action is carried out after the tenth occurrence.

  > **NOTE:** You can set the threshold value to 0 for errors with a severity level of minor. This implies that no action is taken for that error. You cannot set the threshold value to 0 for errors with a severity level of major or fatal.

  Reset and offline are not listed as default actions for the minor error level for safety purposes.

  The alarm and reset default action is included in the implementation for the SRX5000 line of devices. It is required to trigger the RG1 switchover in a chassis cluster environment when an FPC fatal error occurs and the alarm being raised is a major alarm.

  Default: The error count for fatal and major actions is 1. The default error count for minor actions is 10.

  Table 20 on page 312 shows the range of values for each error level.

Table 20: Value Ranges for Error Levels

Error Level	Default Threshold	Value Range
Fatal	1	1 through 1024
Major	1	1 through 1024
Minor	10	0 through 1024

The available detection and recovery actions are as follows:

- **alarm**—Raise an alarm.
- **disable-pfe**—Disable the Packet Forwarding Engine interfaces on the FPC.
- **get-state**—Get the current state of the FPC.
- **log**—Generate a log for the event.
- **offline**—Take the FPC offline.
- **offline-pic**—Take the PIC (installed in the FPC) offline.
- **reset**—Reset the FPC.

Required Privilege Level

interface—To view this in the configuration.

interface-control—To add this to the configuration.

RELATED DOCUMENTATION

fpc error | 325

fin-invalidate-session

Syntax

```
fin-invalidate-session;
```

Hierarchy Level

```
[edit security flow tcp-session]
```

Release Information
Statement introduced in Junos OS Release 10.4 R13.

Description
Invalidates a TCP session after the 4-way or 3-way handshake completes, with each session endpoint signalling conclusion of the session independently. New incoming SYN packets will need to establish a new TCP session.

When either session endpoint wants to terminate the session, it sends a FIN(ish) message. When the other session endpoint receives the packet with the FIN flag set, it sends an ACK(nowlege) message. Typically, tearing down a session involves transmission of a pair of FIN-ACK messages from each session endpoint.

After the side that sent the first FIN responds with the final ACK, it waits for a time-out period to expire before closing the connection. During the time-out period, the local port cannot be used for new connections. The time-out period protects against delayed packets from the terminating session being delivered during subsequent connections.

> **NOTE:** On SRX Series devices with **fin-invalidate-session** configured the invalidation of the session occurs immediately whereas without **fin-invalidate-session** configured the session is set to time out 2 seconds after the 4-way or 3-way handshake completes.

Table 21 on page 315 shows the sequence of packets for a 4-way handshake to terminate a session. In this case, the client signals the server that it is terminating the session. The server responds with an ACK message signaling acknowledgement of the client's FIN message. The ACK is followed immediately by a FIN message that the server sends to the client, signaling that it is terminating the session connection on its end. Finally, the client sends an ACK message to the server signalling that it received the server's FIN message.

Table 21: Terminating a Session with a 4-Way Handshake

Step	Client	Server
1.	FIN	
2.		ACK
3.		FIN • Sets session timer to 150 seconds.
4.	ACK • Sets session timer to 2 seconds.	

A session can be terminated by a 3-way handshake. In this case, the client sends a FIN message to the server. The server responds with message that combines the FIN and ACK messages. The sequence of packet exchange for a three-way handshake session close is as follows:

Step	Client	Server
1.	FIN	
2.		FIN/ ACK • sets session timer to 150 seconds
3.	ACK • sets session timer to 2 seconds	

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

flow (Security Flow)

Syntax

```
flow {
    advanced-options {
        drop-matching-link-local-address;
        drop-matching-reserved-ip-address;
        reverse-route-packet-mode-vr;
    }
    aging {
        early-ageout seconds;
        high-watermark percent;
        low-watermark percent;
    }
    allow-dns-reply;
    allow-embedded-icmp;
    allow-reverse-ecmp;
    enable-reroute-uniform-link-check {
        nat;
    }
    enhanced-routing-mode;
    ethernet-switching {
        block-non-ip-all;
        bpdu-vlan-flooding;
        bypass-non-ip-unicast;
        no-packet-flooding {
            no-trace-route;
        }
    }
    force-ip-reassembly;
    ipsec-performance-acceleration (Security Flow);
    load-distribution {
        session-affinity {
            ipsec;
        }
    }
    mcast-buffer-enhance;
    packet-log (Security Flow) {
        enable;
        packet-filter name {
            conn-tag conn-tag;
            destination-port (afs | bgp | biff | bootpc | bootps | cmd | cvspserver | dhcp | domain | eklogin | ekshell | exec
                | finger | ftp | ftp-data | http | https | ident | imap | kerberos-sec | klogin | kpasswd | krb-prop | krbupdate |
```

```
            kshell | ldap | ldp | login | mobileip-agent | mobilip-mn | msdp | netbios-dgm | netbios-ns | netbios-ssn | nfsd
            | nntp | ntalk | ntp | pop3 | pptp | printer | radacct | radius | range | rip | rkinit | smtp | snmp | snmptrap |
            snpp | socks | ssh | sunrpc | syslog | tacacs | tacacs-ds | talk | telnet | tftp | timed | who | xdmcp | zephyr-clt
            | zephyr-hm | zephyr-srv);
        destination-prefix destination-prefix;
        interface interface;
        logical-system logical-system;
        protocol (ah | egp | esp | gre | icmp | icmp6 | igmp | ipip | number | ospf | pim | rsvp | sctp | tcp | udp);
        source-port (afs | bgp | biff | bootpc | bootps | cmd | cvspserver | dhcp | domain | eklogin | ekshell | exec |
            finger | ftp | ftp-data | http | https | ident | imap | kerberos-sec | klogin | kpasswd | krb-prop | krbupdate |
            kshell | ldap | ldp | login | mobileip-agent | mobilip-mn | msdp | netbios-dgm | netbios-ns | netbios-ssn | nfsd
            | nntp | ntalk | ntp | pop3 | pptp | printer | radacct | radius | range | rip | rkinit | smtp | snmp | snmptrap |
            snpp | socks | ssh | sunrpc | syslog | tacacs | tacacs-ds | talk | telnet | tftp | timed | who | xdmcp | zephyr-clt
            | zephyr-hm | zephyr-srv);
        source-prefix source-prefix;
    }
    throttle-interval milliseconds;
}
pending-sess-queue-length (high | moderate | normal);
power-mode-ipsec;
preserve-incoming-fragment-size;
route-change-timeout seconds;
syn-flood-protection-mode (syn-cookie | syn-proxy);
sync-icmp-session;
tcp-mss (Security Flow) {
    all-tcp {
        mss mss;
    }
    gre-in {
        mss mss;
    }
    gre-out {
        mss mss;
    }
    ipsec-vpn (Security Flow) {
        mss mss;
    }
}
```

```
tcp-session {
    fin-invalidate-session;
    maximum-window (128K | 1M | 256K | 512K | 64K);
    no-sequence-check;
    no-syn-check;
    no-syn-check-in-tunnel;
    rst-invalidate-session;
    rst-sequence-check;
    strict-syn-check;
    tcp-initial-timeout seconds;
    time-wait-state {
        (session-ageout | session-timeout seconds);
        apply-to-half-close-state;
    }
}
traceoptions (Security Flow) {
    file <filename> <files files> <match match> <size size> <(world-readable | no-world-readable)>;
    flag name;
    no-remote-trace;
    packet-filter name {
        conn-tag conn-tag;
        destination-port (afs | bgp | biff | bootpc | bootps | cmd | cvspserver | dhcp | domain | eklogin | ekshell | exec
            | finger | ftp | ftp-data | http | https | ident | imap | kerberos-sec | klogin | kpasswd | krb-prop | krbupdate |
            kshell | ldap | ldp | login | mobileip-agent | mobilip-mn | msdp | netbios-dgm | netbios-ns | netbios-ssn | nfsd
            | nntp | ntalk | ntp | pop3 | pptp | printer | radacct | radius | range | rip | rkinit | smtp | snmp | snmptrap |
            snpp | socks | ssh | sunrpc | syslog | tacacs | tacacs-ds | talk | telnet | tftp | timed | who | xdmcp | zephyr-clt
            | zephyr-hm | zephyr-srv);
        destination-prefix destination-prefix;
        interface interface;
        logical-system logical-system;
        protocol (ah | egp | esp | gre | icmp | icmp6 | igmp | ipip | number | ospf | pim | rsvp | sctp | tcp | udp);
        source-port (afs | bgp | biff | bootpc | bootps | cmd | cvspserver | dhcp | domain | eklogin | ekshell | exec |
            finger | ftp | ftp-data | http | https | ident | imap | kerberos-sec | klogin | kpasswd | krb-prop | krbupdate |
            kshell | ldap | ldp | login | mobileip-agent | mobilip-mn | msdp | netbios-dgm | netbios-ns | netbios-ssn | nfsd
            | nntp | ntalk | ntp | pop3 | pptp | printer | radacct | radius | range | rip | rkinit | smtp | snmp | snmptrap |
            snpp | socks | ssh | sunrpc | syslog | tacacs | tacacs-ds | talk | telnet | tftp | timed | who | xdmcp | zephyr-clt
            | zephyr-hm | zephyr-srv);
        source-prefix source-prefix;
    }
    rate-limit rate-limit;
    trace-level {
        (brief | detail | error);
    }
}
```

```
}
```

Hierarchy Level

```
[edit security]
```

Release Information

Statement modified in Junos OS Release 9.5. The **power-mode-ipsec** option added in Junos OS Release 18.3R1 for vSRX instances, in Junos OS Release 18.4R1 for SRX4100 and SRX4200 devices, and in Junos OS Release 18.2R2 for SRX5400, SRX5600, and SRX5800 devices.

Description

Determine how the device manages packet flow. The device can regulate packet flow in the following ways:

Options

advanced-options— Flow configuration advanced options.

Values:

- close-matching-icmp-session—Allow icmp sessions to be invalidated immediately.
- drop-matching-link-local-address—Drop matching link local address.
- drop-matching-reserved-ip-address—Drop matching reserved source IP address.
- reverse-route-packet-mode-vr—Allow reverse route lookup with packet mode vr.

allow-dns-reply— Allow unmatched incoming DNS reply packet.

allow-embedded-icmp— Allow embedded ICMP packets not matching a session to pass through.

allow-reverse-ecmp— Allow reverse ECMP route lookup.

enable-reroute-uniform-link-check— Enable reroute check with uniform link.

Values:

- nat—Enable NAT check.

enhanced-routing-mode— Enable enhanced route scaling.

force-ip-reassembly— Force to reassemble IP fragments.

ipsec-performance-acceleration— Accelerate the IPSec traffic performance.

mcast-buffer-enhance— Allow to hold more packets during multicast session creation.

pending-sess-queue-length— Maximum queued length per pending session.

Values:

- high—Maximum number of queued sessions.
- moderate—Allow more queued sessions than normal.
- normal—Normal number of sessions queued.

power-mode-ipsec— Enable power mode ipsec processing.

preserve-incoming-fragment-size— Preserve incoming fragment size for egress MTU.

route-change-timeout— Timeout value for route change to nonexistent route (seconds).

Default: 6

Range: 6 through 1800

syn-flood-protection-mode— TCP SYN flood protection mode.

Values:

- syn-cookie—Enable SYN cookie protection.

- syn-proxy—Enable SYN proxy protection.

sync-icmp-session—Allow icmp sessions to sync to peer node.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

Understanding Session Characteristics for SRX Series Services Gateways | 68

Understanding Packet Flow in Logical Systems for SRX Series Devices

force-ip-reassembly

Syntax

```
force-ip-reassembly;
```

Hierarchy Level

```
[edit security flow]
```

Release Information
Statement introduced in Junos OS Release 11.2.

Description
Reassemble all IP fragmented packets before forwarding.

This option is disabled by default. You can disable this option by deleting this flag from the CLI.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

forwarding-process

Syntax

```
forwarding-process {
    application-services (Security Policies) {
        enable-gtpu-distribution;
        maximize-alg-sessions;
        maximize-idp-sessions {
            weight (firewall | idp);
        }
        packet-ordering-mode (Application Services) {
            (hardware |software);
        }
    }
}
```

Hierarchy Level

```
[edit security]
```

Release Information

Statement introduced in Junos OS Release 9.6. This statement is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX instances.

Description

You can configure SRX5400, SRX5600, and SRX5800 devices to switch from an integrated firewall mode to maximize Intrusion Detection and Prevention (IDP) mode to run IDP processing in tap mode and increase the capacity of processing with the **maximize-idp-sessions** option. Inline tap mode can only be configured if the forwarding process mode is set to **maximize-idp-sessions**, which ensures stability and resiliency for firewall services. You also do not need a separate tap or span port to use inline tap mode. When you maximize IDP, you are decoupling IDP processes from firewall processes, allowing the device to support the same number of firewall and IDP sessions, also run the IDP processing in tap mode.

You can configure maximum Application Layer Gateway (ALG) sessions by using the **maximize-alg-sessions** option. By default, the session capacity number for Real-Time Streaming Protocol (RTSP), FTP, and Trivial File Transfer Protocol (TFTP) ALG sessions is 10,000 per flow Services Processing Unit (SPU). You must reboot the device (and its peer in chassis cluster mode) for the configuration to take effect. The **maximize-alg-sessions** option now enables you to increase defaults as follows:

- RTSP, FTP, and TFTP ALG session capacity: 25,000 per flow SPU
- TCP proxy connection capacity: 40,000 per flow SPU

> **NOTE:** Flow session capacity is reduced to half per flow SPU; therefore the aforementioned capacity numbers will not change on central point flow.

Enable GPRS tunneling protocol, user plane(GTP-U) session distribution to distribute GTP-U traffic handled by a Gateway GPRS Support Node (GGSN) and a Serving GPRS Support Node (SGSN) pair on all Services Processing Units (SPUs). You can configure tunnel-base distribution to distribute GTP-U traffic to multiple SPUs by the **enable-gtpu-distribution** option on SRX5400, SRX5600, and SRX5800 devices , which helps to resolve the GTP-U fat session issue. Also, **enable-gtpu-distribution** command is must for enabling stateful GTP-U inspection.

Options

The remaining statements are explained separately. See the CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

application-services (Security Forwarding Process) | 299

Understanding Traffic Processing on Security Devices | 25

fpc error

Syntax

```
fpc slot number {
    error {
        (fatal | major | minor) {
        threshold threshold-value;
        action (alarm | disable-pfe | offline-pic | log | get-state |offline | reset);
        }
    }
}
```

Hierarchy Level

[edit chassis]

Release Information

Statement introduced in Junos OS Release 15.1X49-D40. This command is supported on the SRX5400, SRX5600, and SRX5800 devices.

Description

Configure the threshold at which FPC errors will take the action you configure to be performed by the device.

Some devices include an internal framework for detecting and correcting FPC errors that can have the potential to affect services. For each FPC on the device, you can classify errors according to severity, set an automatic recovery action for each severity, and set a threshold (the number of times the error must occur before the action is triggered).

Options

You can configure the threshold for the following severity levels:

> **NOTE:** You cannot change the severity level of an error.

- **fatal**—Fatal error on the FPC. An error that results in the blockage of a considerable amount of traffic across modules is a fatal error.
- **major**—Major error on the FPC. An error that results in continuing loss of packet traffic but does not affect other modules is a major error.

- **minor**—Minor error on the FPC. An error that results in the loss of a single packet but is fully recoverable is a minor error.
- **threshold *threshold-value***—Configure the threshold value at which to take action. If the severity level of the error is fatal, the action is carried out only once when the total number of errors exceeds the threshold value. If the severity level of the error is major, the action is carried out once after the occurrence exceeds the threshold. If the severity level is minor, the action is carried out as many times as the value specified by the threshold. For example, when the severity level is minor, and you have configured the threshold value as 10, the action is carried out after the tenth occurrence.

  NOTE: You can set the threshold value to 0 for errors with a severity level of minor. This implies that no action is taken for that error. You cannot set the threshold value to 0 for errors with a severity level of major or fatal.

  Default: The error count for fatal and major actions is 1. The default error count for minor actions is 10.

  Range: 0—429,496,729

The available detection and recovery actions are as follows:

- **alarm**—Raise an alarm.
- **disable-pfe**—Disable the Packet Forwarding Engine interfaces on the FPC.
- **get-state**—Get the current state of the FPC.
- **log**—Generate a log for the event.
- **offline**—Take the FPC offline.
- **offline-pic**—Take the PIC (installed in the FPC) offline.
- **reset**—Reset the FPC.

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

RELATED DOCUMENTATION

error | 310

fru-poweron-sequence

Syntax

```
fru-poweron-sequence fru-poweron-sequence;
```

Hierarchy Level

```
[edit chassis]
```

Release Information
Statement introduced in Junos OS Release 12.1X44-D10.

Description
Configure the power-on sequence for FPCs installed in the chassis. SRX5400, SRX5600, and SRX5800 devices.

Options
fru-poweron-sequence—Power-on sequence for the FPCs in the chassis. The numbers indicate the slot number of the FPCs.

> **NOTE:** If the power-on sequence is not configured by including the **fru-poweron-sequence** , Junos OS uses the ascending order of the the FPC slot numbers as the sequence for powering on the FPCs.

> **NOTE:** The FPC online sequence is not dependent on the FPC power-on sequence.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

gre-in

Syntax

```
gre-in {
    mss value;
}
```

Hierarchy Level

```
[edit security flow tcp-mss]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Enable and specify the TCP maximum segment size (TCP MSS) for Generic Routing Encapsulation (GRE) packets that are coming out from an IPsec VPN tunnel. If the device receives a GRE-encapsulated TCP packet with the SYN bit and TCP MSS option set and the TCP MSS option specified in the packet exceeds the TCP MSS specified by the device, the device modifies the TCP MSS value accordingly. By default, a TCP MSS for GRE packets is not set.

Options

mss *value* —TCP MSS for GRE packets. Value is optional.

Range: 64 through 63535 bytes

Default: 1320 bytes, if no value is specified

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

gre-out

Syntax

```
gre-out {
    mss value;
}
```

Hierarchy Level

```
[edit security flow tcp-mss]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Enable and specify the TCP maximum segment size (TCP MSS) for Generic Routing Encapsulation (GRE) packets that are going into an IPsec VPN tunnel. If the device receives a GRE-encapsulated TCP packet with the SYN bit and TCP MSS option set and the TCP MSS option specified in the packet exceeds the TCP MSS specified by the device, the device modifies the TCP MSS value accordingly. By default, a TCP MSS for GRE packets is not set.

Options

mss *value*—TCP MSS for GRE packets. Value is optional.

Range: 64 through 65,535 bytes

Default: 1320 bytes

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

high-watermark

Syntax

```
high-watermark percent;
```

Hierarchy Level

```
[edit security flow aging]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Sets the point at which the aggressive aging-out process begins.

Options

percent —Percentage of session-table capacity at which aggressive aging-out starts.

Range: 1 through 100 percent

Default: 100 percent

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

hop-by-hop-header

Syntax

```
hop-by-hop-header {
    CALIPSO-option;
    RPL-option;
    SFM-DPD-option;
    jumbo-payload-option;
    quick-start-option;
    router-alert-option;
    user-defined-option-type low | <to high>;
}
```

Hierarchy Level

```
[edit security screen ids-option screen-name ip ipv6-extension-header]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Define the IPv6 hop-by-hop screen option.

Options

CALIPSO-option—Enable the Common Architecture Label IPv6 Security Option.

RPL-option—Enable the Routing Protocol for Low-Power and Lossy Networks screen option.

SMF-DPD-option—Enable the Simplified Muliticast Forwarding IPv6 Duplicate Packet Detection screen option.

jumbo-payload-option—Enable the IPv6 jumbo payload screen option.

quick-start-option—Enable the IPv6 quick start screen option.

router-alert-option—Enable the IPv6 router alert screen option.

user-defined-header-type *low* | <to *high*>—Define the type of header range.

Range: 1 through 255.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding IPv6 Support for Screens

ipv6-extension-header | 338

destination-header | 302

icmpv6-malformed

Syntax

```
icmpv6-malformed;
```

Hierarchy Level

[edit security screen ids-option *screen-name* icmp]

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Enable the ICMPv6 malformed intrusion detection service (IDS) option.

Options

This has no options.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding IPv6 Support for Screens

ipv6-extension-header | 338

idle-timeout (System Services)

Syntax

```
idle-timeout;
```

Hierarchy Level

```
[edit system services web-management session]
```

Release Information
Statement introduced in Junos OS Release 9.0.

Description
Configure in minutes the idle-timeout parameter for web-management sessions. The idle-timeout parameter, which applies to all sessions, specifies the length of time a session can be idle before it is terminated. The web-management feature allows you to configure the device using the J-Web interface.

You can also configure the maximum allowed number of concurrent web management sessions using the session-limit parameter.

Options
idle-timeout—*minutes*

Default: 30

Range: 1 to 1440

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

RELATED DOCUMENTATION

Firewall User Authentication Overview

Dynamic VPN Overview

inline-tap

Syntax

```
inline-tap;
```

Hierarchy Level

```
[edit security forwarding-process application-services maximize-idp-sessions]
```

Release Information
Statement introduced in Junos OS Release 10.2.

Description
Enable IDP inline tap mode. The inline tap feature provides passive, inline detection of application layer threats for traffic matching security policies which have the IDP application service enabled. When a device is in inline tap mode, packets pass through firewall inspection and are also copied to the independent IDP module.

This statement is supported in SRX1500, SRX 5800, SRX 5600, and SRX 5400 devices and vSRX.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

interface-in (Security Forwarding Options)

Syntax

```
interface-in interface-name;
```

Hierarchy Level

```
[edit security forwarding-options mirror-filter filter-name]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Specify the incoming logical interface to be matched for mirroring.

Required Privilege Level

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

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346

show security forwarding-options mirror-filter | 736

interface-out (Security Forwarding Options)

Syntax

```
interface-out interface-name;
```

Hierarchy Level

```
[edit security forwarding-options mirror-filter filter-name]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Specify the outgoing logical interface to be matched for mirroring.

Required Privilege Level

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

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346
show security forwarding-options mirror-filter | 736

ipv4-template (Services)

Syntax

```
ipv4-template;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name]
```

Release Information

Statement introduced in Junos OS Release 10.4.

Description

Specify that the flow monitoring version 9 template is used only for IPv4 records.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

Understanding Interfaces

ipv6-extension-header

Syntax

```
ipv6-extension-header {
    AH-header;
    ESP-header
    HIP-header;
    destination-header {
        ILNP-nonce-option;
        home-address-option;
        line-identification-option;
        tunnel-encapsulation-limit-option;
        user-defined-option-type low | <to high>;
    }
    fragment-header;
    hop-by-hop-header {
        CALIPSO-option;
        RPL-option;
        SFM-DPD-option;
        jumbo-payload-option;
        quick-start-option;
        router-alert-option;
        user-defined-option-type low | <to high>;
    }
    mobility-header;
    no-next-header;
    routing-header;
    shim6-header
    user-defined-option-type low | <to high>;
}
```

Hierarchy Level

```
[edit security screen ids-option screen-name ip]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Define the IPv6 extension header for the intrusion detection service (IDS).

Options

AH-header—Enable the IPv6 Authentication Header screen option.

ESP-header—Enable the IPv6 Encapsulating Security Payload header screen option.

HIP-header—Enable the IPv6 Host Identify Protocol header screen option.

fragment-header—Enable the IPv6 fragment header screen option.

mobility-header—Enable the IPv6 mobility header screen option.

no-next-header—Enable the IPv6 no next header screen option.

routing-header—Enable the IPv6 routing header screen option.

shim6-header—Enable the IPv6 shim header screen option.

user-defined-header-type *low* | <to *high*>—Define the type of header range.

Range: 0 through 255.

The remaining statements are explained separately. See the CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding IPv6 Support for Screens

hop-by-hop-header | 331

destination-header | 302

ipv6-extension-header-limit

Syntax

```
ipv6-extension-header-limit limit;
```

Hierarchy Level

```
[edit security screen ids-option screen-name ip]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Define the IPv6 extension header number limit for screen options. The screen blocks packets that have more than the defined number of extension headers.

Options

limit—Set the number of IPv6 extension headers that can pass through the screen.

Range: 0 through 32.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding IPv6 Support for Screens

ipv6-extension-header | 338

ipv6-malformed-header

Syntax

```
ipv6-malformed-header;
```

Hierarchy Level

```
[edit security screen ids-option screen-name ip]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Enable the IPv6 malformed header intrusion detection service (IDS) option.

Options

This has no options.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding IPv6 Support for Screens

ipv6-extension-header | 338

ipv6-template (Services)

Syntax

```
ipv6-template;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name]
```

Release Information

Statement introduced in Junos OS Release 12.1X45-D10.

Description

Specify that the flow monitoring version 9 template is used only for IPv6 records.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

Understanding Interfaces

low-latency

Syntax

```
low-latency
```

Hierarchy Level

```
[edit chassis fpc fpc-slot-number pic  pic-slot-number services-offload]
```

Release Information

Statement introduced in Junos OS Release 12.1X44-D10.

Description

Enables the low-latency mode on the selected NP-IOC. Low-latency is not enabled by default. The low latency allows you to configure the mode of the network processor's traffic manager (TM) on the egress path. If low latency is enabled, the network processor is initialized without the traffic manager, thus reducing the overall latency in the Express Path (formerly known as services offloading).

> **NOTE:** Because all SRX Series CoS functions are supported by the traffic manager, CoS functions are not supported when low latency is enabled.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
Example: Configuring Low Latency | 284

low-watermark

Syntax

```
low-watermark percent;
```

Hierarchy Level

```
[edit security flow aging]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Set the point at which the aggressive aging-out process ends.

Options

percent —Percentage of session-table capacity at which aggressive aging-out ends.

Range: 0 through 100 percent

Default: 100 percent

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

maximize-idp-sessions

Syntax

```
maximize-idp-sessions {
    weight (Security) (equal | firewall | idp);
}
```

Hierarchy Level

```
[edit security forwarding-process application-services]
```

Release Information

Statement introduced in Junos OS Release 9.6.

Description

If you are deploying IDP policies, you can tune the device to increase IDP session capacity. By using the provided commands to change the way the system allocates resources, you can achieve a higher IDP session capacity. See weight for information about the options provided.

This statement is supported on SRX1500, SRX 5800, SRX 5600, and SRX 5400 devices and vSRX.

NOTE: The IDP session capacity is restricted to 100,000 sessions per SPU.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

mirror-filter (Security Forwarding Options)

Syntax

```
mirror-filter filter-name {
    destination-port port-number;
    destination-prefix destination-prefix;
    interface-in interface-name;
    interface-out interface-name;
    output (Security Forwarding Options) {
        destination-mac mac-address;
        interface interface-name;
    }
    protocol protocol;
    source-port port-number;
    source-prefix set source-prefix;
}
```

Hierarchy Level

[edit security forwarding-options]

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Configure a mirror filter for filtering X2 packets to be mirrored and sent to a packet analyzer.

As a network operator, you need a way to monitor X2 traffic to debug any handover issues across eNodeBs. The mirror filter feature allows you to do that. Traffic coming out of an IPsec tunnel is decrypted, mirrored and analyzed, and then encrypted again to go into the outbound IPsec tunnel.

To use the mirror filter feature to monitor X2 traffic, you configure mirror filters. You can configure up to 15 different mirror filters to be used concurrently to filter for various kinds of traffic. Each mirror filter contains a set of parameters and their values against which traffic is matched.

NOTE: The SRX Series mirror filter feature is bidirectional, much like a session. X2 traffic flowing through an IPSec VPN from devices that match the configured filter conditions is mirrored and analyzed.

Starting in Junos OS Release 18.4R1, if the output X2 interface of a mirror filter is configured for an st0 interface to filter traffic that you want to analyze, the packet is duplicated and encrypted by the IPsec tunnel bound to the st0 interface. This enhancement supports the SRX Series devices to send traffic mirrored from a port on an IPsec tunnel.

In addition to the following parameters for a mirror filter, you specify the output interface and the MAC address of the packet analyzer as part of the configuration.

> **NOTE:** Although there is no minimum required number of parameters for a mirror filter, please be mindful that if you specify too few criteria or accidentally commit an incomplete filter, an over-proportional amount of traffic flow through the system could be mirrored.

- destination IP address prefix
- destination port
- IP protocol
- source IP address prefix
- source port
- incoming and outgoing interfaces

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

show security forwarding-options mirror-filter | 736

clear security forward-options mirror filter | 450

mode (Security Forwarding Options)

Syntax

```
mode (drop | flow-based | packet-based);
```

Hierarchy Level

```
[edit security forwarding-options family inet6]
```

Release Information

Support on SRX Series devices for flow-based mode for family inet6 added in Junos OS Release 10.2.

Description

Specify forwarding options for IPv6 traffic.

Options

- **drop**—Drop IPv6 packets. This is the default setting.
- **flow-based**—Perform flow-based packet forwarding.
- **packet-based**—Perform simple packet forwarding.

> **NOTE:** Packet-based processing is not supported on the following SRX Series devices: SRX5400, SRX5600, and SRX5800.

- Starting with Junos OS Release 15.1X49-D70, on SRX1500, SRX4100, SRX4200, SRX5400, SRX5600, SRX5800 devices, the default mode is changed from drop mode to flow based mode.
- If you change the forwarding option mode for IPv6, you might have to perform a reboot to initialize the configuration change. Table 22 on page 349 summarizes device status upon configuration change.
- Starting with Junos OS Release 15.1X49-D70, on SRX5400, SRX5600, and SRX5800 devices, reboot is not required when you change the modes between flow-based mode and drop mode.

Table 22: Device Status Upon Configuration Change

Configuration Change	Commit Warning	Reboot Required	Impact on Existing Traffic Before Reboot	Impact on New Traffic Before Reboot
Drop to flow-based	Yes	Yes	Dropped	Dropped
Drop to packet-based	No	No	Packet-based	Packet-based
Flow-based to packet-based	Yes	Yes	None	Flow sessions created
Flow-based to drop	Yes	Yes	None	Flow sessions created
Packet-based to flow-based	Yes	Yes	Packet-based	Packet-based
Packet-based to drop	No	No	Dropped	Dropped

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

no-sequence-check

Syntax

```
no-sequence-check;
```

Hierarchy Level

```
[edit security flow tcp-session]
```

Release Information
Statement introduced in Junos OS Release 8.5.

Description
Specify that the device does not check sequence numbers in TCP segments during stateful inspection. By default, the device monitors the sequence numbers in TCP segments. The device detects the window scale specified by source and destination hosts in a session and adjusts a window for an acceptable range of sequence numbers according to their specified parameters. The device then monitors the sequence numbers in packets sent between these hosts. If the device detects a sequence number outside this range, it drops the packet.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

np-cache (Flexible PIC Concentrator)

Syntax

```
np-cache;
```

Hierarchy Level

```
[edit chassis fpc fpc-slot-number]
```

Release Information

Statement introduced in Junos OS Release 15.1X49-D10.

Description

Enable session cache table on IOC.

Starting with Junos OS Release 15.1X49-D10 and and Junos OS Release 17.3R1, NP cache is supported on the SRx5K-MPC (IOC2), SRX5K-MPC3-100G10G (IOC3), and SRX5K-MPC3-40G10G (IOC3) for SRX5400, SRX5600, and SRX5800 devices.

The security policy determines whether a session is for Express Path (formerly known as *services offloading*) mode on the selected Flexible PIC Concentrator (FPC).

NOTE: The IOC2 and the IOC3 utilize the delay sessions delete mechanism. The same sessions (sessions with the same five tuples) that are deleted and then reinstalled immediately are not cached on the IOCs.

NOTE: To enable IPsec VPN affinity, you must also enable the session cache on IOCs (IOC2 and IOC3) by using the **set chassis fpc <*fpc-slot*> np-cache** command.

Required Privilege Level

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

RELATED DOCUMENTATION

Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path | 272

Example: Configuring SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) on an SRX5000 Line Device to Support Express Path | 275

output (Security Forwarding Options)

Syntax

```
output {
    destination-mac mac-address;
    interface interface-name;
}
```

Hierarchy Level

[edit security forwarding-options mirror-filter *filter-name*]

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Specify the MAC address or interface for mirrored traffic.

Starting in Junos OS Release 18.4R1, if the output X2 interface of a mirror filter is configured for an st0 interface to filter traffic that you want to analyze, the packet is duplicated and encrypted by the IPsec tunnel bound to the st0 interface. This enhancement supports the SRX Series devices to send traffic mirrored from a port on an IPsec tunnel.

Options

destination-mac *mac-address*—Specify the MAC address for the mirrored traffic.

interface *interface-name*—Specify the logical interface for the mirrored traffic.

Required Privilege Level

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

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346

show security forwarding-options mirror-filter | 736

packet-filter

Syntax

```
packet-filter packet-filter-name {
    action-profile profile-name {
    destination-port (Security Forwarding Options) (port-range | protocol-name);
    destination-prefix destination-prefix;
    interface logical-interface-name;
    protocol (Security Forwarding Options) (protocol-number | protocol-name;
    source-port (Security Forwarding Options) (port-range | protocol- name);
    source-prefix source-prefix;
    }
```

Hierarchy Level

```
[edit security datapath-debug]
```

Release Information

Command introduced in Junos OS Release 9.6 ; Support for IPv6 addresses for the **destination-prefix** and **source-prefix** options added in Junos OS Release 10.4.

Description

Set packet filter for taking the datapath-debug action. A filter is defined to filter traffic, then an action profile is applied to the filtered traffic. Be sure to configure multiple packet filters to capture the traffic. One packet filter only captures the traffic as specified in it, such as from one source to one destination. The same packet filter will not capture the traffic in the reverse direction. You need to configure another packet filter to capture the traffic in reverse direction and specify the source and destination according to the response packet in it. The action profile specifies a variety of actions on the processing unit. A maximum of four filters are supported at the same time. Packet filters can be configured with source and destination prefix and port (including ranges), and protocol.

Action-profile settings have no specific minimum setting, it is based on trace, count, packet summary and packet-dump. Enabling end-to-end debugging without or with a very broad filter is not recommended. This could result in a high PFE CPU usage. Therefore when selecting what to capture through a filter care must be taken. List as many and specific criteria which then results in the minimum amount of traffic to be captured.

NOTE: Packet filter is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800 devices.

Options

- **action-profile *profile-name***—Identify the action profile to use. You can specify the name of the action profile to use. Using the request security action-profile command, you can set the action for the packet match for a specified filter. Action-profile must be defined.
- **destination-port (*port-range* | *protocol name*)**—Specify a destination port to match TCP/UDP destination port.
- **destination-prefix *destination-prefix***—Specify a destination IPv4/IPv6 address prefix.
- **interface *logical-interface-name***—Specify a logical interface name.
- **protocol (*protocol-number* | *protocol-name***—Match IP protocol type.
- **source-port (*port-range* | *protocol-name***—Match TCP/UDP source port.
- **source-prefix *source-prefix***—Specify a source IP address prefix.

Required Privilege Level

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

packet-log (Security Flow)

Syntax

```
packet-log {
    enable;
    throttle-interval;
    packet-filter <filter-name>;
}
```

Hierarchy Level

```
[edit security flow ]
```

Release Information
Statement introduced in Junos OS Release 17.3R1.

Description
Configure flow packet log.

Starting in Junos OS Release 17.3R1, SRX Series devices support logging of unsynchronized out-of-state TCP packets that are dropped by the flow module.

The logging functionality will allow notifications to the administrators (or automated systems) about any of TCP out-of-state packets drops.

Options
enable—Enable log for dropped packets.

packet-filter—Configure packet log filter.

throttle-interval—Configure the interval as a power of two (0..32768 milliseconds).

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

packet-ordering-mode (Application Services)

Syntax

```
packet-ordering-mode {
    (hardware | software);
}
```

Hierarchy Level

[edit security forwarding-process application-services]

Release Information

Statement introduced in Junos OS Release 12.1X45-D10. This statement is supported on the SRX5400, SRX5600, and SRX5800 devices and vSRX.

Description

Enables or disables the packet-ordering functionality using the Packet Ordering Engine. By default, packet-ordering functionality using the Packet Ordering Engine (hardware) is enabled.

A system reboot is required when this feature is enabled or disabled, and a warning message is displayed during the commit.

> **NOTE:** Packet-ordering functionality using Packet Ordering Engine is supported on SRX5800 and SRX5600 devices with next-generation SPCs. Starting from Junos OS release 12.1X46-D10, SRX5400 device with next-generation SPCs also supports this feature.

Options

hardware— Enables packet-ordering functionality using the Packet Ordering Engine.

software— Disables packet-ordering functionality using the Packet Ordering Engine.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

pending-sess-queue-length

Syntax

```
pending-sess-queue-length (high | moderate | normal);
```

Hierarchy Level

```
[edit security flow]
```

Release Information

Statement introduced in Junos OS Release 11.4.

Description

Configure the maximum queued length per pending session.

Options

- **high**— Allow the maximum number of queued sessions.
- **moderate**—Allow more queued sessions than the normal number.
- **normal**—Allow the normal number of queued session.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

per-session-statistics

Syntax

```
per-session-statistics;
```

Hierarchy Level

```
[edit chassis fpc name name pic name services-offload]
```

Release Information
Statement introduced in Junos OS Release 11.4.

Description
Keeps per session statistics in NPC monitoring.

In services offload, once the session is established, packet no longer goes to Services Processing Unit (SPU). The Network Processing Card (NPC) will send the packet statistics periodically (1 second interval) to SPU. This means in services offload session, NPC will store packet statistics on per session basis.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
services-offload | 371

pre-id-default-policy

Syntax

```
pre-id-default-policy {
    then {
        log {
            session-init;
            session-close;
        }
        session-timeout {
            icmp seconds;
            icmp6 seconds;
            ospf seconds;
            others seconds;
            tcp seconds;
            udp seconds;
        }
    }
}
```

Hierarchy Level

```
[edit security policies]
```

Release Information

Statement introduced in Junos OS Release 18.2R1.

Description

During the initial policy lookup phase, which occurs prior to a dynamic application being identified, if there are multiple policies present in the potential policy list, the SRX Series device applies the default security policy until a more explicit match has occurred. Configures default policy actions that occur prior to dynamic application identification (AppID).

CAUTION: Configuring **session-init** logging for the **pre-id-default-policy** can generate a large amount of logs. Each session that enters the SRX that initially matches the **pre-id-default-policy** will generate an event. We recommend only using this option for troubleshooting purposes.

Options

then—Specifies the policy action that has to be taken when the packet matches the criteria.

log— Specifies the log details at session close time and session initialization time.

Values:

- session-init—Log at the beginning of a session
- session-close—Log at the closure of a session

> **NOTE:** In recent versions of Junos OS, the factory-default configuration of an SRX includes the **session-close** knob. If not already present, we recommend that customers implement **session-close logging** within their **pre-id-default-policy**. This will ensure security logs are generated by the SRX if a flow is unable to leave the **pre-id-default-policy**. These events are generally a result of JDPI being unable to properly classify traffic, although they may also indicate potential attempts at evading the APPID engine.

session-timeout—When you update a session, the session timeout is configured, which specifies the session timeout details in seconds.

Values: icmp—Timeout value for ICMP sessions (seconds)

Range: 4 through 86,400

Values: icmp6—Timeout value for ICMP6 sessions (seconds)

Range: 4 through 86,400

Values: ospf—Timeout value for OSPF sessions (seconds)

Range: 4 through 86,400

Values: others—Timeout value for other sessions (seconds)

Range: 4 through 86,400

Values: tcp—Timeout value for TCP sessions (seconds)

Range: 4 through 86,400

Values: udp—Timeout value for UDP sessions (seconds)

Range: 4 through 86,400

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Advanced Policy-Based Routing

preserve-incoming-fragment-size

Syntax

```
preserve-incoming-fragment-size;
```

Hierarchy Level

```
[edit security flow] ]
```

Release Information
Statement introduced in Junos OS Release 15.1X49-D100.

Description
Enable the preserve incoming fragment size feature that allows the SRX Series device to preserve the size of incoming fragments to be used in determining the best maximum transmission unit (MTU) size for the egress interface.

When data is sent from one host to another, it is transmitted as a series of packets. Performance is improved and network resources are conserved when packets of the largest size can transit the path from the source node to the destination node without being fragmented at any link in the datapath.

If application-layer services are configured on the SRX Series device, packet fragments at the ingress interface must be reassembled before the services can be applied and the content inspected. These reassembled packet fragments must be broken down again before the data is transmitted out the egress interface.

When a packet must be fragmented into smaller packets to transit a link in the path because the packet is larger than the MTU size established for that link, each of the resulting fragments must contain packet header information, in addition to the payload, or data. The increased overhead can lower throughput and degrade network performance. Also, the packet fragments must be reassembled at the destination node, which consumes additional network resources.

By default, the SRX Series device uses the MTU size configured for the egress interface to determine the size for the packet fragments it transmits. However, if you enable the preserve incoming fragment size feature, the SRX Series device detects and saves the size of incoming packet fragments and takes that into account. To diminish the likelihood of packet fragmentation in the datapath, the SRX Series device sets the egress interface MTU size to the smaller of two values: It identifies the maximum size of all incoming fragments and it compares that size to the existing MTU size of the egress interface. The SRX Series device takes the smaller number and uses it for the egress interface MTU size.

Required Privilege Level
security—To view this statement in the configuration.

security-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding How Preserving Incoming Fragmentation Characteristics Can Improve Throughput | 84

propagate-settings

Syntax

```
propagate-settings interface-name;
```

Hierarchy Level

```
[edit system services dhcp]
[edit system services dhcp pool]
```

Release Information
Statement introduced in Junos OS Release 8.5.

Description
Enable or disable the propagation of TCP/IP settings received on the device acting as Dynamic Host Configuration Protocol (DHCP) client. The settings can be propagated to the server pool running on the device. Use the **system services dhcp** to set this feature globally. Use the **system services dhcp pool** to set the feature for the address pool and override the global setting.

Options
logical-interface-name —Name of the logical interface to receive TCP/IP settings from the external network for propagation to the DHCP pool running on the device.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

protocol (Security Forwarding Options)

Syntax

```
protocol protocol;
```

Hierarchy Level

[edit security forwarding-options mirror-filter *filter-name*]

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Specify the networking protocol name or number to be matched for mirroring.

Options

protocol-name—Networking protocol name or number. The following text values are supported. For a complete list of possible numeric values, see RFC 1700, *Assigned Numbers (for the Internet Protocol Suite)*.

ah—IP Security Authentication header

egp—Exterior gateway protocol

esp—IPsec Encapsulating Security Payload

gre—Generic routing encapsulation

icmp—Internet Control Message Protocol

icmp6—Internet Control Message Protocol version 6

igmp—Internet Group Management Protocol

ipip—IP over IP

ospf—Open Shortest Path First

pim—Protocol Independent Multicast

rsvp—Resource Reservation Protocol

sctp—Stream Control Transmission Protocol

tcp—Transmission Control Protocol

udp—User Datagram Protocol

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346

show security forwarding-options mirror-filter | 736

resource-manager

Syntax

```
resource-manager {
    traceoptions (Security) {
        flag flag;
    }
}
```

Hierarchy Level

```
[edit security]
```

Release Information

Statement introduced before Junos OS Release 12.1.

Description

Configure resource manager security options.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

reverse-route-packet-mode-vr

Syntax

```
reverse-route-packet-mode-vr;
```

Hierarchy Level

```
[edit security flow advanced-options]
```

Release Information
Statement introduced in Junos OS Release 18.2R1.

Description
When processing a server to client traffic, if the route is changed, the server to client traffic is rerouted. The traffic is rerouted using the virtual router from the first incoming session interface. The command allows reverse route lookup with packet mode using virtual router.

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

RELATED DOCUMENTATION

Understanding Packet-Based Processing | 211

route-change-timeout

Syntax

```
route-change-timeout seconds;
```

Hierarchy Level

```
[edit security flow]
```

Release Information

Statement introduced in Junos OS Release 8.5. Support for default value set to 6 seconds added in Junos OS Release 12.1X45-D10.

Description

Specify the session timeout when a session is rerouted but there is a reroute failure (for example, the new route uses a different egress zone from the previous route).

Options

seconds —Amount of time before sessions are timed out.

Range: 6 through 1800 seconds

Default: 6 seconds

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

rst-invalidate-session

Syntax

```
rst-invalidate-session;
```

Hierarchy Level

```
[edit security flow tcp-session]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Enable the device to mark a session for immediate termination when it receives a TCP reset (RST) message. By default, this feature is disabled.

> **NOTE:** On SRX Series devices with **rst-invalidate-session** configured the invalidation of the session occurs immediately whereas without **rst-invalidate-session** configured the session is set to time out 2 seconds after a TCP reset (RST) message has been received.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

rst-sequence-check

Syntax

```
rst-sequence-check;
```

Hierarchy Level

```
[edit security flow tcp-session]
```

Release Information
Statement introduced in Junos OS Release 8.5.

Description
Verify that the TCP sequence number in a TCP segment with the RST bit enabled matches the previous sequence number for a packet in that session or is the next higher number incrementally. If the sequence number does not match either of these expected numbers, the device drops the packet and sends the host a TCP ACK message with the correct sequence number. By default, this check is disabled.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

sampling

Syntax

```
sampling {
    command binary-file-path;
    disable;
    failover (alternate-media | other-routing-engine);
}
```

Hierarchy Level

```
[edit system processes]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Perform packet sampling based on particular input interfaces and various fields in the packet header.

Options

- **command *binary-file-path***—Path to the binary process.
- **disable**—Disable the traffic sampling control process.
- **failover**—Configure the device to reboot if the software process fails four times within 30 seconds, and specify the software to use during the reboot.
  - **alternate-media**—Configure the device to switch to backup media that contains a version of the system if a software process fails repeatedly.
  - **other-routing-engine**—Instruct the secondary Routing Engine to take mastership if a software process fails. If this is configured for a process, and that process fails four times within 30 seconds, then the device reboots from the secondary Routing Engine.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

services-offload

Syntax

```
services-offload {
    low-latency;
    per-session-statistics;
}
```

Hierarchy Level

[edit chassis fpc ***fpc-slot-number*** pic ***pic-slot-number***]

Release Information

Statement introduced in Junos OS Release 11.4.

Description

Enables the Express Path mode (formerly known as *services offloading*) mode on the selected network processor. Services-offload is not enabled by default.

When services-offload is enabled, only the first packets of a session goes to the Services Processing Unit (SPU), rest of packets in services-offload mode does not go to SPU, therefore some security features such as stateful screen are not supported. Only TCP and UDP packets can be services offloaded.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

session (System Services)

Syntax

```
session {
    idle-timout minutes;
    session-limit number;
}
```

Hierarchy Level

```
[edit system services web-management]
```

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Configure parameters for web-management sessions. Web management allows for configuration of the SRX Series device using the J-Web interface. You can configure the idle-timeout parameter for web-management sessions and the maximum number of concurrent sessions.

Options

idle-timeout *minutes*—Configure in minutes the time-out parameter for all web-management sessions. The idle-timeout parameter specifies the length of time that a session can be idle before it is terminated.

Default: 1440

Range: 1 to 1440

session-limit *number*—Configure the maximum allowed number of concurrent web management sessions. By default, an unlimited number of users can log in to the J-Web interface on a Juniper Networks device, and each session remains open for 24 hours (1440 minutes).

Default: unlimited

Range: 1 to 1024

Required Privilege Level

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

RELATED DOCUMENTATION

Firewall User Authentication Overview

Dynamic VPN Overview

session-limit (System Services)

Syntax

```
session-limit;
```

Hierarchy Level

```
[edit system services web-management session]
```

Release Information
Statement introduced in Junos OS Release 9.0.

Description
Configure the maximum allowed number of concurrent web management sessions. Using the CLI, you can limit the number of concurrent sessions from 1 to 1024. Each session remains open for 24 hours (1440 minutes).

You can also set an idle time-out parameter to override the default to specify the length of time a session can be idle before it is terminated.

Options
session-limit—*number*

Default: unlimited sessions

Range: 1 to 1024 sessions

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

RELATED DOCUMENTATION

Firewall User Authentication Overview

Dynamic VPN Overview

source-port (Security Forwarding Options)

Syntax

```
source-port port-number;
```

Hierarchy Level

```
[edit security forwarding-options mirror-filter filter-name]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D10.

Description

Specify a Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) source port number to be matched for mirroring. You can specify a numeric value or one of the text synonyms listed in Table 23 on page 375.

Table 23: Ports Supported by Services Interfaces

Port Name	Corresponding Port Number
afs	1483
bgp	179
biff	512
bootpc	68
bootps	67
cmd	514
cvspserver	2401
dhcp	67
domain	53
eklogin	2105
ekshell	2106
excc	512
finger	79
ftp	21
ftp-data	20
http	80
https	443
ident	113
imap	143
kerberos-sec	88
klogin	543

Table 23: Ports Supported by Services Interfaces *(continued)*

Port Name	Corresponding Port Number
kpasswd	761
krb-prop	754
krbupdate	760
kshell	544
ldap	389
ldp	646
login	513
mobileip-agent	434
mobilip-mn	435
msdp	639
netbios-dgm	138
netbios-ns	137
netbios-ssn	139
nfsd	2049
nntp	119
ntalk	518
ntp	123
pop3	110
pptp	1723
printer	515
radacct	1813

Table 23: Ports Supported by Services Interfaces *(continued)*

Port Name	Corresponding Port Number
radius	1812
rip	520
rkinit	2108
smtp	25
snmp	161
snmp-trap	162
snpp	444
socks	1080
ssh	22
sunrpc	111
syslog	514
tacacs	49
tacacs-ds	65
talk	517
telnet	23
tftp	69
timed	525
who	513
xdmcp	177

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346
show security forwarding-options mirror-filter | 736

source-prefix (Security Forwarding Options)

Syntax

```
source-prefix source-prefix;
```

Hierarchy Level

[edit security forwarding-options mirror-filter *filter-name*]

Release Information
Statement introduced in Junos OS Release 12.1X46-D10.

Description
Specify the source IP prefix or address to be matched for mirroring.

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

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346
show security forwarding-options mirror-filter | 736

syn-flood-protection-mode

Syntax

```
syn-flood-protection-mode (syn-cookie | syn-proxy);
```

Hierarchy Level

```
[edit security flow]
```

Release Information
Statement introduced in Junos OS Release 8.5; support for IPv6 addresses added in Junos OS Release 10.4.

Description
Enable SYN cookie or SYN proxy defenses against SYN attacks. SYN flood protection mode is enabled globally on the device and is activated when the configured **syn-flood attack-threshold value** is exceeded.

Options

- **syn-cookie**—Uses a cryptographic hash to generate a unique Initial Sequence Number (ISN). This is enabled by default.
- **syn-proxy**—Uses a proxy to handle the SYN attack.

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

tcp-initial-timeout

Syntax

```
tcp-initial-timeout seconds;
```

Hierarchy Level

```
[edit security flow tcp-session]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Define the length of time (in seconds) that the device keeps an initial TCP session in the session table before dropping it, or until the device receives a FIN (no more data) or RST (reset) packet. The FIN flag indicates the end of data transmission to finish a TCP connection.

Options

seconds—Number of seconds that the device keeps an initial TCP session in the session table before dropping it.

Range: 4 through 300 seconds

Default: 20 seconds

> **NOTE:** The minimum value you can configure for TCP session initialization is 4 seconds. The default value is 20 seconds; if required you can set the TCP session initialization value to less than 20 seconds.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

tcp-mss (Security Flow)

Syntax

```
tcp-mss {
    all-tcp mss value;
    gre-in {
        mss value;
    }
    gre-out {
        mss value;
    }
    ipsec-vpn {
        mss value;
    }
}
```

Hierarchy Level

```
[edit security flow]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Configure TCP maximum segment size (TCP MSS) for the following packet types:

- All TCP packets for network traffic.
- GRE packets entering the IPsec VPN tunnel.
- GRE packets exiting the IPsec VPN tunnel.
- TCP packets entering the IPsec VPN tunnel.

If all the four TCP MSS options are configured simultaneously, then the order of preference is as follows:

- If TCP packet enters an IPsec VPN tunnel, then an ipsec-vpn mss value has high priority over all-tcp mss value, hence ipsec-vpn mss value is set.
- If TCP packet enters GRE , then gre-in mss value overrides all-tcp mss value, hence gre-in mss value is set.
- If TCP packet exits GRE, then all-tcp mss value overrides gre-in mss value, hence all-tcp mss value is set.

Options

The remaining statements are explained separately. See CLI Explorer.

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

RELATED DOCUMENTATION

- all-tcp | **295**
- gre-in | **328**
- gre-out | **329**
- *ipsec-vpn (Security Flow)*
- Understanding Traffic Processing on Security Devices | **25**

tcp-session

Syntax

```
tcp-session {
    no-sequence-check;
    no-syn-check;
    no-syn-check-in-tunnel;
    rst-invalidate-session;
    rst-sequence-check;
    strict-syn-check;
    tcp-initial-timeout seconds;
    time-wait-state {
        (session-ageout | session-timeout seconds);
    }
}
```

Hierarchy Level

```
[edit security flow]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

Configure TCP session attributes:

- TCP sequence number checking.
- TCP SYN bit checking.
- Reset (RST) checking.
- Initial TCP session timeout—The minimum value you can configure for TCP session initialization is 4 seconds. The default value is 20 seconds; if required you can set the TCP session initialization value to less than 20 seconds.
- Strict TCP SYN checking.
- TCP session timeout for time-wait state.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

security—To view this in the configuration.

security-control—To add this to the configuration.

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

time-wait-state

Syntax

```
time-wait-state {
    (session-ageout | session-timeout seconds);
}
```

Hierarchy Level

```
[edit security flow tcp-session]
```

Release Information

Statement introduced in Junos OS Release 11.1.

Description

Defines the length of time (in seconds) that the device keeps the defined TCP session in the session table. The default is 150 seconds.

Options

- **session-ageout**—Set a TCP session to age out, using the service based timeout value.
- **session-timeout *seconds***—Set the session timeout value allowed before the device ages out a session from its session table.

  Range: 2 through 600 seconds

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

traceoptions (Security)

Syntax

```
traceoptions {
    file {
        filename;
        files number;
        match regular-expression;
        size maximum-file-size;
        (world-readable | no-world-readable);
    }
    flag flag;
    no-remote-trace;
    rate-limit messages-per-second;
}
```

Hierarchy Level

```
[edit security]
```

Release Information

Statement modified in Junos OS Release 8.5.

Description

Configure security tracing options.

Options

- **file**—Configure the trace file options.
  - ***filename***—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory **/var/log**. By default, the name of the file is the name of the process being traced.
  - **files *number***—Maximum number of trace files. When a trace file named ***trace-file*** reaches its maximum size, it is renamed to ***trace-file*.0**, then ***trace-file*.1**, and so on, until the maximum number of trace files is reached. The oldest archived file is overwritten.

    If you specify a maximum number of files, you also must specify a maximum file size with the **size** option and a filename.

    Range: 2 through 1000 files

    Default: 10 files
  - **match *regular-expression***—Refine the output to include lines that contain the regular expression.

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

  If you specify a maximum file size, you also must specify a maximum number of trace files with the **files** option and a filename.

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

  Range: 10 KB through 1 GB

  Default: 128 KB

- **world-readable** | **no-world-readable**—By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.

- **flag**—Trace operation to perform. To specify more than one trace operation, include multiple **flag** statements.
  - **all**—Trace all security events
  - **compilation**—Trace security compilation events
  - **configuration**—Trace security configuration events
  - **routing-socket**—Trace routing socket events
- **no-remote-trace**—Set remote tracing as disabled.
- **rate-limit *messages-per-second***—Limit the incoming rate of trace messages.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

traceoptions (Security Flow)

Syntax

```
traceoptions {
    file {
        filename;
        files number;
        match regular-expression;
        size maximum-file-size;
        (world-readable | no-world-readable);
    }
    flag flag;
    no-remote-trace;
    packet-filter filter-name {
        conn-tag session-conn
        destination-port port-identifier;
        destination-prefix address;
        interface interface-name;
        protocol protocol-identifier;
        source-port port-identifier;
        source-prefix address;
    }
    rate-limit messages-per-second;
    trace-level (brief | detail | error);
}
```

Hierarchy Level

```
[edit security flow]
[edit logical-systems logical-system-name security flow]
[edit tenants tenant-name security flow]
```

Release Information

Statement introduced in Junos OS Release 8.5.
Statement updated in Junos OS Release 12.1X46-D10 with the **trace-level** option and additional flags.
Statement updated in Junos OS Release 15.1X49-D70 with the addition of the **conn-tag** filter parameter.
Support at the following hierarchy levels introduced in Junos OS Release 19.4R1: **[edit logical-systems *logical-system-name* security flow]**, and **[edit tenants *tenant-name* security flow]**.

Description

Configure flow tracing options.

Options

file—Configure the trace file options.

filename—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory **/var/log**. By default, the name of the file is the name of the process being traced.

files *number*—Maximum number of trace files. When a trace file named ***trace-file*** reaches its maximum size, it is renamed to ***trace-file*.0**, then ***trace-file*.1**, and so on, until the maximum number of trace files is reached. The oldest archived file is overwritten.

If you specify a maximum number of files, you also must specify a maximum file size with the **size** option and a filename.

Range: 2 through 1000 files

Default: 10 files

match *regular-expression*—Refine the output to include lines that contain the regular expression.

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

If you specify a maximum file size, you also must specify a maximum number of trace files with the **files** option and a filename.

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

Range: 0 KB through 1 GB

Default: 128 KB

world-readable | no-world-readable—By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.

flag—Trace operation to perform. To specify more than one trace operation, include multiple **flag** statements.

all—Trace with all flags enabled

basic-datapath—Trace basic packet flow activity

fragmentation—Trace IP fragmentation and reassembly events

high-availability—Trace flow high-availability information

host-traffic—Trace flow host traffic information

multicast—Trace multicast flow information

route—Trace route lookup information

session—Trace session creation and deletion events

session-scan—Trace session scan information

tcp-basic—Trace TCP packet flow information

tunnel—Trace tunnel information

no-remote-trace—Set remote tracing as disabled.

packet-filter *filter-name*—Packet filter to enable during the tracing operation. Configure the filtering options.

destination-port *port-identifier*—Match TCP/UDP destination port

destination-prefix *address*—Destination IP address prefix

interface *interface-name*—Logical interface

protocol *protocol-identifier*—Match IP protocol type

source-port *port-identifier*—Match TCP/UDP source port

source-prefix *address*—Source IP address prefix

rate-limit *messages-per-second*—Limit the incoming rate of trace messages.

trace-level—Set the level for trace logging. This option is available only when the flag is set.

brief—Trace key flow information, such as message types sent between SPU and central point, policy match, and packet drop reasons.

detail—Trace extensive flow information, such as detailed information about sessions and fragments. Detail is the default level.

error—Trace error information, such as system failure, unknown message type, and packet drop.

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

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

transport (Security Log)

Syntax

```
transport {
    protocol (udp | tcp | tls);
    tls-profile tls-profile-name;
    tcp-connections tcp-connections;
}
```

Hierarchy Level

```
[edit security log]
[edit logical-systems name security log]
[edit tenants tenant-name security log]
```

Release Information

Statement introduced in Junos OS Release 12.1X46-D25.
The [edit **logical-systems** *name* security log] and [edit **tenants** *tenant-name* security log] hierarchy levels introduced in Junos OS Release 19.1R1.

Description

Configure security log transport options.

Options

protocol—Specify the type of transport protocol to be used to log the data.

- **UDP**—Set the transport protocol to UDP.
- **TCP**—Set the transport protocol to TCP.
- **TLS**—Set the transport protocol to TLS.

  Default: UDP.

tls-profile *tls-profile-name*—Specify the TLS profile name.

tcp-connections *tcp-connections*—Specify the number of TCP connections per SPU.

Range: 1 through 5.

Default: 1.

Required Privilege Level

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

RELATED DOCUMENTATION

Understanding AppTrack

weight (Security)

Syntax

```
weight (equal | firewall | idp);
```

Hierarchy Level

```
[edit security forwarding-process application-services maximize-idp-sessions]
```

Release Information
Statement introduced in Junos OS Release 9.6.

Description
If you are deploying IDP policies, you can tune the device to increase IDP session capacity. By using the provided commands to change the way the system allocates resources, you can achieve a higher IDP session capacity.

Devices ship with an implicit default session capacity setting. This default value gives more weight to firewall sessions. You can manually override the default by using the **maximize-idp-sessions** command. The command allows you to choose between these weight values: **equal**, **firewall**, and **idp**. The following table displays the available session capacity weight and approximate throughput for each.

Table 24: Session Capacity and Resulting Throughput

Weight Value	Firewall Capacity	IDP Capacity	Firewall Throughput	IDP Throughput
Default	1,000,000	256,000	10 Gbps	2.4 Gbps
equal	1,000,000	1,000,000	8.5 Gbps	2 Gbps
firewall	1,000,000	1,000,000	10 Gbps	2.4 Gbps
idp	1,000,000	1,000,000	5.5 Gbps	1.4 Gbps

This statement is supported on SRX1500, SRX 5800, SRX 5600, and SRX 5400 devices and vSRX.

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

RELATED DOCUMENTATION

CHAPTER 7

Operational Commands

clear firewall | 399

clear monitor security flow filter | 401

clear security flow ip-action | 402

clear security flow session all | 405

clear security flow session application | 407

clear security flow session application-traffic-control | 410

clear security flow session conn-tag | 413

clear security flow session destination-port | 415

clear security flow session destination-prefix | 417

clear security flow session family | 419

clear security flow session IDP | 421

clear security flow session interface | 424

clear security flow ip-action | 426

clear security flow session nat | 429

clear security flow session protocol | 432

clear security flow session resource-manager | 435

clear security flow session services-offload | 437

clear security flow session session-identifier | 441

clear security flow session source-port | 443

clear security flow session source-prefix | 445

clear security flow session tunnel | 447

clear security forward-options mirror filter | 450

monitor security flow file | 451

monitor security flow filter | 453

monitor security flow start | 455

monitor security flow stop | 456

show chassis environment (Security) | 457

show chassis fpc (View) | 464

show chassis fpc errors | 475

show chassis hardware (View) | 478

show chassis pic (Security) | 499

show chassis power | 502

show chassis power sequence | 507

show firewall (View) | 509

show interfaces (View Aggregated Ethernet) | 512

show interfaces diagnostics optics | 526

show interfaces flow-statistics | 532

show interfaces swfabx | 538

show monitor security flow | 540

show security flow cp-session | 542

show security flow cp-session destination-port | 546

show security flow cp-session destination-prefix | 549

show security flow cp-session family | 552

show security flow cp-session protocol | 555

show security flow cp-session source-port | 558

show security flow cp-session source-prefix | 561

show security flow gate | 564

show security flow ip-action | 569

show security flow gate brief node | 580

show security flow gate destination-port | 588

show security flow gate destination-prefix | 592

show security flow gate protocol | 596

show security flow gate summary node | 600

show security flow pmi statistics | 607

show security flow session | 610

show security flow session brief node | 620

show security flow session destination-port | 625

show security flow session destination-prefix | 630

show security flow session extensive node | 636

show security flow session family | 644

show security flow session interface | 651

show security flow session nat | 656

show security flow session policy-id | 660

show security flow session protocol | 664

show security flow session resource-manager | 670

show security flow session services-offload | 675

show security flow session session-identifier | 682

show security flow session source-port | 688

show security flow session source-prefix | 693

show security flow session summary family | 698

show security flow session summary node | 701

show security flow session summary services-offload | 709

show security flow session tunnel | 714

show security flow statistics | **725**

show security flow status | **731**

show security forwarding-options mirror-filter | **736**

show security monitoring | **739**

show security policies | **742**

show security policies hit-count | **759**

show security resource-manager group active | **763**

show security resource-manager resource active | **767**

show security resource-manager settings | **771**

show security resource-manager summary | **774**

show security screen ids-option | **776**

show security screen statistics | **783**

show security softwires | **797**

show security zones | **799**

show security zones type | **807**

clear firewall

Syntax

```
clear firewall
<all>
<counter counter-name>
<filter filter-name>
```

Release Information

Command introduced in Junos OS Release 10.0.

Description

Clear statistics about configured firewall filters.

> **NOTE:** The **clear firewall** command cannot be used to clear the Routing Engine filter counters on a backup Routing Engine that is enabled for GRES.

If you clear statistics for firewall filters that are applied to Trio-based DPCs and that also use the **prefix-action** action on matched packets, wait at least 5 seconds before you enter the **show firewall prefix-action-stats** command. A 5-second pause between issuing the **clear firewall** and **show firewall prefix-action-stats** commands avoids a possible timeout of the **show firewall prefix-action-stats** command.

Options

all—Clear the packet and byte counts for all filters.

counter *counter-name*—Clear the packet and byte counts for a filter counter that has been configured with the counter firewall filter action.

filter *filter-name*—Clear the packet and byte counts for the specified firewall filter.

Required Privilege Level

clear

RELATED DOCUMENTATION

show firewall (View) | 509

List of Sample Output

clear firewall all on page 400

Output Fields

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

Sample Output

clear firewall all

```
user@host> clear firewall all
```

clear monitor security flow filter

Syntax

```
clear monitor security flow filter <filter-name>
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10.

Description

Specify the security flow filters to be deleted. Once deleted, the filters are removed from the Packet Forwarding Engine and the Routing Engine. .

> **NOTE:** Specifying the filter name is optional. If no filter is specified, all filters are deleted.

Options

This command has no options.

Required Privilege Level

clear

RELATED DOCUMENTATION

- Monitoring Security Flow Sessions Overview | 186
- monitor security flow start | 455
- monitor security flow filter | 453

Output Fields

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

Sample Output

clear security flow ip-action

Syntax

```
clear security flow ip-action [filter]
```

Release Information
Command introduced in Junos OS Release 10.4. Logical systems option introduced in Junos OS Release 11.2.

Description
Clear IP-action entries, based on filtered options, for IP sessions running on the device.

Options
filter—Filter the display based on the specified criteria.

The following filters display those sessions that match the criteria specified by the filter. Refer to the sample output for filtered output examples.

all | [*filter*]—All active sessions on the device.

destination-port *destination-port*—Destination port number of the traffic. Range is 1 through 65,535.

destination-prefix *destination-prefix*—Destination IP prefix or address.

family (inet | inet6) [*filter*]—IPv4 traffic or IPv6-NATPT traffic and filtered options.

logical-system *logical-system-name* | all [*filter*]—Specified logical system or all logical systems.

protocol *protocol-name* | *protocol-number* [*filter*]—Protocol name or number and filtered options.

- **ah** or **51**
- **egp** or **8**
- **esp** or **50**
- **gre** or **47**
- **icmp** or **1**
- **icmp6** or **58**
- **ipip** or **4**
- **ospf** or **89**
- **pim** or **103**
- **rsvp** or **46**

- **sctp** or **132**
- **tcp** or **6**
- **udp** or **17**

root-logical-system [*filter*]—Default logical system information and filtered options.

source-port ***source-port***—Source port number of the traffic. Range is 1 through 65,535.

source-prefix ***source-prefix***—Source IP prefix or address of the traffic.

Required Privilege Level
clear

RELATED DOCUMENTATION

show security flow ip-action | 569

List of Sample Output
clear security flow ip-action all on page 403
clear security flow ip-action destination-prefix on page 403
clear security flow ip-action family inet on page 404
clear security flow ip-action protocol udp on page 404

Output Fields
When you enter this command, the system responds with the status of your request.

Sample Output

clear security flow ip-action all

`user@host>`**clear security flow ip-action all**

```
1008 ip-action entries cleared
```

clear security flow ip-action destination-prefix

`user@host>`**clear security flow ip-action destination-prefix 192.0.2.5/24**

```
87 ip-action entries cleared
```

clear security flow ip-action family inet

```
user@host>clear security flow ip-action family inet
```

```
2479 ip-action entries cleared
```

clear security flow ip-action protocol udp

```
user@host>clear security flow ip-action protocol udp
```

```
270 ip-action entries cleared
```

clear security flow session all

Syntax

```
clear security flow session all
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5 ; **node** options added in Junos OS Release 9.0.

Description

Clear all currently active security sessions on the device.

Options

- all—Clear information about all active sessions.
- **node**—(Optional) For chassis cluster configurations, clear all security sessions on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session | **610**

List of Sample Output

clear security flow session all on page 406
clear security flow session all node 0 on page 406

Output Fields

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

Sample Output

clear security flow session all

user@host> **clear security flow session all**

```
node0:
--------------------------------------------------------------------------
1 active sessions cleared
node1:
-------- ----------------------------------------------------------------
0 active sessions cleared
```

Sample Output

clear security flow session all node 0

user@host> **clear security flow session all node 0**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session application

Syntax

```
clear security flow session application
application-name
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5. The **node** options added in Junos OS Release 9.0.

Description

Clear currently active sessions for application types or application sets.

Options

- ***application-name*** —Name of the specified application type or application set.
  - **dns**—Domain Name System
  - **ftp**—File Transfer Protocol
  - **ignore**—Ignore application type
  - **mgcp-ca**—Media Gateway Control Protocol with Call Agent
  - **mgcp-ua**—MGCP with User Agent
  - **ms-rpc**—Microsoft RPC
  - **pptp**—Point-to-Point Tunneling Protocol
  - **q931**—ISDN connection control protocol
  - **ras**—RAS
  - **realaudio**—RealAudio
  - **rsh**—UNIX remote shell services
  - **rtsp**—Real-Time Streaming Protocol
  - **sccp**—Skinny Client Control Protocol
  - **sip**—Session Initiation Protocol
  - **sqlnet-v2**—Oracle SQLNET
  - **sun-rpc**—Sun Microsystems RPC

- **talk**—TALK program
- **tftp**—Trivial File Transfer Protocol

- **node**—(Optional) For chassis cluster configurations, clear sessions for applications on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level
clear

RELATED DOCUMENTATION

show security flow session application

List of Sample Output
clear security flow session application dns on page 408
clear security flow session application dns node 0 on page 409

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

Sample Output

clear security flow session application dns

`user@host>` **clear security flow session application dns**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

Sample Output

clear security flow session application dns node 0

```
user@host> clear security flow session application dns node 0
```

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session application-traffic-control

Syntax

clear security flow session application-traffic-control [Enter] | *application-traffic-control-session-options*

Release Information

Command introduced in Junos OS Release 11.4.

Description

Clear all application traffic control sessions or the session associated with the specified option.

Application traffic control enables application-aware quality of service, as specified in the rules of the rules set defined for it.

Options

application—Name of the application or application set.

- **dns**—Domain Name System
- **ftp**—File Transfer Protocol
- **ignore**—Ignore application type
- **mgcp-ca**—Media Gateway Control Protocol with Call Agent
- **mgcp-ua**—MGCP with User Agent
- **ms-rpc**—Microsoft RPC
- **pptp**—Point-to-Point Tunneling Protocol
- **q931**—ISDN connection control protocol
- **ras**—RAS
- **realaudio**—RealAudio
- **rsh**—UNIX remote shell services
- **rtsp**—Real-Time Streaming Protocol
- **sccp**—Skinny Client Control Protocol
- **sip**—Session Initiation Protocol
- **sqlnet-v2**—Oracle SQLNET
- **sun-rpc**—Sun Microsystems RPC
- **talk**—TALK program
- **tftp**—Trivial File Transfer Protocol

To display the supported applications list on an SRX Series device, enter the following command from configuration mode:

```
user@host# show groups junos-default applications
```

conn-tag—A 32-bit connection tag that uniquely identifies the GPRS tunneling protocol, user plane (GTP-U), and the Stream Control Transmission Protocol (STCP) sessions.

The connection tag for GTP-U is the tunnel endpoint identifier (TEID). For SCTP, it is the vTag. The connection ID remains 0 if the connection tag is not used by the sessions.

Range: 0 through 4294967295

destination-port—Destination port.

Range: 1 through 65535.

destination-prefix—Destination IP prefix or address.

family— Protocol family:

- inet—Clear IPv4 sessions.
- inet6–Clear IPv6 sessions.

interface—Name of incoming or outgoing interface.

protocol —IP protocol number.

source-port— Source port.

Range: 1 through 65535.

source-prefix—Source IP prefix or address.

Required Privilege Level
clear

RELATED DOCUMENTATION

show security flow session | 610

Output Fields

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

The same output is displayed when you enter an option for this command.

Sample Output

```
user@host> clear security flow session application-traffic-control

number active sessions cleared
```

clear security flow session conn-tag

Syntax

```
clear security flow session conn-tag conn-tag-id-number
```

Release Information

Command introduced in Junos OS Release 15.1X49-D40.

Description

Clear the session identified by the session connection (conn-tag) identification tag.

The flow session connection tag allows you to add a filter to further distinguish GRSP tunneling protocol, user plane (GTP-U) flow sessions, and Stream Control Transmission Protocol (SCTP) flow sessions.

Options

conn-tag—A 32-bit connection tag that uniquely identifies the GPRS tunneling protocol, user plane (GTP-U) and the Stream Control Transmission Protocol (STCP) sessions. The connection tag for GTP-U is the tunnel endpoint identifier (TEID) and for SCTP is the vTag. The connection ID remains 0 if the connection tag is not used by the sessions.

You can configure the system to include the session connection tag tuple to identify GTP-U session and SCTP sessions by adding the session connection tag to the standard six tuples that identify a session. The system determines the DCP for GTP-U/SCTP by hashing the session connection tag.

The central point architecture distributes GTP-U traffic handled by a gateway GPRS support node (GGSN) and SGSN pair on all SPUs by switching to tunnel endpoint identifier (TEID)-based hash distribution. To handle load-balancing issues, tag-based hash distribution is used to ensure even distribution of SCTP traffic from different associations among all SPUs. (The connection tag for GTP-U is the TEID and for SCTP is the vTag.)

Range: 0 through 4294967295.

Required Privilege Level

clear

RELATED DOCUMENTATION

show-security-flow

Output Fields

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

Sample Output

user@host> **clear-security-flow-session conn-tag *conn-tag-id-number***

```
number active sessions cleared
```

clear security flow session destination-port

Syntax

```
clear security flow session destination-port
destination-port-number
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5 ; **node** options added in Junos OS Release 9.0.

Description

Clear each session that uses the specified destination port

Options

- ***destination-port-number*** —Number of the destination port.
- **node**—(Optional) For chassis cluster configurations, clear security sessions on the port on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session destination-port | 625

List of Sample Output

clear security flow session destination-port 1 on page 416
clear security flow session destination-port 1 node 0 on page 416

Output Fields

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

Sample Output

clear security flow session destination-port 1

user@host> **clear security flow session destination-port 1**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

Sample Output

clear security flow session destination-port 1 node 0

user@host> **clear security flow session destination-port 1 node 0**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session destination-prefix

Syntax

```
clear security flow session destination-prefix
destination-IP-prefix
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Release 8.5 of Junos OS.
The **node** options added in Release 9.0 of Junos OS.
Support for IPv6 addresses added in Release 10.2 of Junos OS.

Description

Clear sessions that match this destination IP prefix or address.

Options

- ***destination-IP-prefix***—Destination IP prefix or address.
- **node**—(Optional) For chassis cluster configurations, clear sessions that match the IP prefix or address on a specific node (device) in the cluster.
  - ***node-id***—Identification number of the node. It can be 0 or 1.
  - **all**—Clear all nodes.
  - **local**—Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session destination-prefix | 630

List of Sample Output

clear security flow session destination-prefix 100.0.0.1 on page 418
clear security flow session destination-prefix 10::10 on page 418
Clear security flow session destination-prefix 100.0.0.1 node 0 on page 418

Output Fields

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

Sample Output

clear security flow session destination-prefix 100.0.0.1

clear security flow session destination-prefix 100.0.0.1

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session destination-prefix 10::10

```
user@host> clear security flow session destination-prefix 10::10
```

```
1 active sessions cleared
```

Sample Output

Clear security flow session destination-prefix 100.0.0.1 node 0

```
user@host> clear security flow session destination-prefix 100.0.0.1 node 0
```

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session family

Syntax

```
clear security flow session family (inet | inet6)
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Clear sessions that match the specified protocol family.

Options

- **inet**—Clear IPv4 sessions.
- **inet6**—Clear IPv6 sessions.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session family | 644

List of Sample Output

clear security flow session family inet on page 419
clear security flow session family inet6 on page 420

Output Fields

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

Sample Output

clear security flow session family inet

```
user@host> clear security flow session family inet

1 active sessions cleared
```

clear security flow session family inet6

```
user@host> clear security flow session family inet6
```

```
1 active sessions cleared
```

clear security flow session IDP

Syntax

```
clear security flow session IDP [Enter] | idp-session-parameters
```

Release Information
Command introduced in Junos OS Release 10.2.

Description
Clear all active Intrusion Detection and Prevention (IDP) sessions or an IDP session based on the specified session parameter.

IDP allows you to enforce various attack detection and prevention techniques on network traffic passing through the SRX Series device. The SRX Series offers the same set of IDP signatures that are available on Juniper Networks IDP Series Intrusion Detection and Prevention Appliances to secure networks against attacks.

Options
application—Name of the application or application set.

- **dns**—Domain Name System
- **ftp**—File Transfer Protocol
- **ignore**—Ignore application type
- **mgcp-ca**—Media Gateway Control Protocol with Call Agent
- **mgcp-ua**—MGCP with User Agent
- **ms-rpc**—Microsoft RPC
- **pptp**—Point-to-Point Tunneling Protocol
- **q931**—ISDN connection control protocol
- **ras**—RAS
- **realaudio**—RealAudio
- **rsh**—UNIX remote shell services
- **rtsp**—Real-Time Streaming Protocol
- **sccp**—Skinny Client Control Protocol
- **sip**—Session Initiation Protocol
- **sqlnet-v2**—Oracle SQLNET
- **sun-rpc**—Sun Microsystems RPC

- **talk**—TALK program
- **tftp**—Trivial File Transfer Protocol

To display the supported applications list on an SRX Series device, enter the following command from configuration mode:

```
user@host# show groups junos-default applications
```

conn-tag—A 32-bit connection tag that uniquely identifies the GPRS tunneling protocol, user plane (GTP-U), and the Stream Control Transmission Protocol (STCP) sessions.

The connection tag for GTP-U is the tunnel endpoint identifier (TEID). For SCTP, it is the vTag. The connection ID remains 0 if the connection tag is not used by the sessions.

Range: 0 through 4294967295

destination-port—Destination port.

Range: 1 through 65535.

destination-prefix—Destination IP prefix or IP address.

family— Protocol family:

- inet—Clear IPv4 sessions.
- inet6–Clear IPv6 sessions.

interface—Name of incoming or outgoing interface.

protocol —IP protocol number.

source-port— Source port.

Range: 1 through 65535.

source-prefix—Source IP prefix or address.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session | 610

Output Fields

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

The same output is displayed when you enter an option for this command.

Sample Output

```
user@host> clear security flow session idp

number active sessions cleared
```

clear security flow session interface

Syntax

```
clear security flow session interface
interface-name
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5; **node** options added in Junos OS Release 9.0.

Description

Clear sessions that use the specified interface.

Options

- ***interface-name*** —Name of a specific incoming or outgoing interface.
- **node**—(Optional) For chassis cluster configurations, clear security sessions on the interface on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session interface | 651

List of Sample Output

clear security flow session interface ge-0/0/0.0 on page 425
clear security flow session interface ge/0/0.0 node 0 on page 425

Output Fields

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

Sample Output

clear security flow session interface ge-0/0/0.0

```
user@host> clear security flow session interface ge-0/0/0.0

node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

Sample Output

clear security flow session interface ge/0/0.0 node 0

```
user@host> clear security flow session interface ge-0/0/0.0 node 0

node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow ip-action

Syntax

clear security flow ip-action [*filter*]

Release Information
Command introduced in Junos OS Release 10.4. Logical systems option introduced in Junos OS Release 11.2.

Description
Clear IP-action entries, based on filtered options, for IP sessions running on the device.

Options
filter—Filter the display based on the specified criteria.

The following filters display those sessions that match the criteria specified by the filter. Refer to the sample output for filtered output examples.

all | [*filter*]—All active sessions on the device.

destination-port *destination-port*—Destination port number of the traffic. Range is 1 through 65,535.

destination-prefix *destination-prefix*—Destination IP prefix or address.

family (inet | inet6) [*filter*]—IPv4 traffic or IPv6-NATPT traffic and filtered options.

logical-system *logical-system-name* | all [*filter*]—Specified logical system or all logical systems.

protocol *protocol-name* | *protocol-number* [*filter*]—Protocol name or number and filtered options.

- **ah** or **51**
- **egp** or **8**
- **esp** or **50**
- **gre** or **47**
- **icmp** or **1**
- **icmp6** or **58**
- **ipip** or **4**
- **ospf** or **89**
- **pim** or **103**
- **rsvp** or **46**

- **sctp** or **132**
- **tcp** or **6**
- **udp** or **17**

root-logical-system [*filter*]—Default logical system information and filtered options.

source-port ***source-port***—Source port number of the traffic. Range is 1 through 65,535.

source-prefix ***source-prefix***—Source IP prefix or address of the traffic.

Required Privilege Level
clear

RELATED DOCUMENTATION

show security flow ip-action | 569

List of Sample Output
clear security flow ip-action all on page 427
clear security flow ip-action destination-prefix on page 427
clear security flow ip-action family inet on page 428
clear security flow ip-action protocol udp on page 428

Output Fields
When you enter this command, the system responds with the status of your request.

Sample Output

clear security flow ip-action all

`user@host>`**clear security flow ip-action all**

```
1008 ip-action entries cleared
```

clear security flow ip-action destination-prefix

`user@host>`**clear security flow ip-action destination-prefix 192.0.2.5/24**

```
87 ip-action entries cleared
```

clear security flow ip-action family inet

user@host>**clear security flow ip-action family inet**

```
2479 ip-action entries cleared
```

clear security flow ip-action protocol udp

user@host>**clear security flow ip-action protocol udp**

```
270 ip-action entries cleared
```

clear security flow session nat

Syntax

```
clear security flow session nat <Enter> | nat-session-parameter
```

Release Information
Command introduced in Junos OS Release 10.2.

Description
Clear all active sessions with NAT configurations or the active NAT session identified by a session parameter.

Network Address Translation (NAT) is now used primarily to improve traffic security. But it also offers solutions to IP address constraints prior to the advent and implementation of IPv6. NAT allows you to remap one IP address space to another. Network address information in the IP datagram packet headers are modified to achieve the translation.

Options
application—Name of the application protocol

- **dns**—Domain Name System
- **ftp**—File Transfer Protocol
- **ignore**—Ignore application type
- **mgcp-ca**—Media Gateway Control Protocol with Call Agent
- **mgcp-ua**—MGCP with User Agent
- **ms-rpc**—Microsoft RPC
- **pptp**—Point-to-Point Tunneling Protocol
- **q931**—ISDN connection control protocol
- **ras**—RAS
- **realaudio**—RealAudio
- **rsh**—UNIX remote shell services
- **rtsp**—Real-Time Streaming Protocol
- **sccp**—Skinny Client Control Protocol
- **sip**—Session Initiation Protocol
- **sqlnet-v2**—Oracle SQLNET
- **sun-rpc**—Sun Microsystems RPC

- **talk**—TALK program
- **tftp**—Trivial File Transfer Protocol

To display a list of the supported applications on an SRX Series device, enter the following command from configuration mode:

```
user@host# show groups junos-default applications
```

conn-tag—Session identified by the specified conn-tag.

A conn-tag is a 32-bit connection tag that uniquely identifies the GPRS tunneling protocol, user plane (GTP-U), and the Stream Control Transmission Protocol (STCP) sessions.

The connection tag for GTP-U is the tunnel endpoint identifier (TEID). For SCTP, it is the vTag. The connection ID remains 0 if the connection tag is not used by the sessions.

Session connection identifiers are in the following range:

Range: 0 through 4294967295

destination-port—Destination port.

Range: 1 through 65535

destination-prefix—Destination IP prefix or address.

family— Protocol family:

- inet—Clear IPv4 sessions
- inet6–Clear IPv6 sessions

interface—Name of incoming or outgoing interface.

protocol —IP protocol number.

source-port— Source port.

Range: 1 through 65535.

source-prefix—Source IP prefix or IP address.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session | 610

Output Fields

Displays a message reporting the number of active sessions cleared. The same message is displayed when any specific option is entered.

Sample Output

`user@host>` **clear security flow session nat**

```
number active sessions cleared
```

clear security flow session protocol

Syntax

```
clear security flow session protocol  protocol-name  |  protocol-number
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5 ; **node** options added in Junos OS Release 9.0.

Description

Clear each session that uses the specified IP protocol.

Options

- ***protocol-name*** — (Optional) Networking protocol name. The following text values are supported.
  - **ah**—IP Security Authentication Header
  - **egp**—Exterior gateway protocol
  - **esp**—IPsec Encapsulating Security Payload
  - **gre**—Generic routing encapsulation
  - **icmp**—Internet Control Message Protocol
  - **igmp**—Internet Group Management Protocol
  - **ipip**—IP over IP
  - **ospf**—Open Shortest Path First
  - **pim**—Protocol Independent Multicast
  - **rsvp**—Resource Reservation Protocol
  - **sctp**—Stream Control Transmission Protocol
  - **tcp**—Transmission Control Protocol
  - **udp**—User Datagram Protocol
- ***protocol-number*** —(Optional) Numeric protocol value. For a complete list of possible numeric values, see RFC 1700, *Assigned Numbers (for the Internet Protocol Suite)*.

  Range: 0 through 255
- **node**—(Optional) For chassis cluster configurations, clear security on a specific node (device) in the cluster for the user with this identification number.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.

- **local** —Clear the local node.
- **primary**—Clear the primary node.

Required Privilege Level
clear

RELATED DOCUMENTATION

show security flow session protocol | 664

List of Sample Output
clear security flow session protocol pim on page 433
clear security flow session protocol 0 on page 433

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

Sample Output

clear security flow session protocol pim

```
user@host> clear security flow session protocol pim
```

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

Sample Output

clear security flow session protocol 0

```
user@host> clear security flow session protocol 0
```

```
node0:
--------------------------------------------------------------------------
```

```
0 active sessions cleared
node1:
--------------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session resource-manager

Syntax

```
clear security flow session resource-manager
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5; **node** options added in Junos OS Release 9.0.

Description

Clear resource-manager sessions.

Options

- **node**—(Optional) For chassis cluster configurations, clear the resource manager sessions on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session resource-manager | 670

List of Sample Output

clear security flow session resource-manager on page 436
clear security flow session resource-manager node 0 on page 436

Output Fields

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

Sample Output

clear security flow session resource-manager

user@host> **clear security flow session resource-manager**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

Sample Output

clear security flow session resource-manager node 0

user@host> **clear security flow session resource-manager node 0**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session services-offload

Syntax

clear security flow session services-offload [*filter*]

Release Information

Command introduced in Junos OS Release 11.4.
Starting with Junos OS Release 15.1X49-D10 and and Junos OS Release 17.3R1, the SRX5K-MPC3-100G10G (IOC3) and the SRX5K-MPC3-40G10G (IOC3) with Express Path (formerly known as *services offloading*) support are introduced for SRX5400, SRX5600, and SRX5800 devices.

Description

Clear services-offload security sessions, based on filtered options, on the device. This command also clears a services-offload security session from both the network processor and the Services Processing Unit (SPU) on which the specified session was installed.

Options

filter—Filter the display based on the specified criteria.

The following filters clear those sessions that match the criteria specified by the filter. Refer to the sample output for filtered output examples.

application *application-name*—Application name.

destination-port *destination-port*—Destination port number. Range is from 1 through 65,535.

destination-prefix *destination-prefix*—Destination IP prefix or address.

family (inet | inet6)—IPv4 traffic or IPv6-NAT-PT traffic.

interface *interface-name*—Incoming or outgoing interface name.

logical-system *logical-system-name* | all—Specified logical system name or all logical systems.

protocol *protocol-name* | *protocol-number* —Protocol name or number.

- **ah** or **51**
- **egp** or **8**
- **esp** or **50**
- **gre** or **47**
- **icmp** or **1**
- **icmp6** or **58**

- **igmp** or **2**
- **ipip** or **4**
- **ospf** or **89**
- **pim** or **103**
- **rsvp** or **46**
- **sctp** or **132**
- **tcp** or **6**
- **udp** or **17**

root-logical-system [*filter*]—Root logical system information and filtered options.

source-port ***source-port***—Source port number of the traffic. Range is from 1 through 65,535.

source-prefix ***source-prefix***—Source IP prefix or address of the traffic.

Required Privilege Level
clear

RELATED DOCUMENTATION

show security flow session services-offload | 675

List of Sample Output

clear security flow session services-offload on page 439
clear security flow session services-offload application on page 439
clear security flow session services-offload destination-port on page 439
clear security flow session services-offload destination-prefix on page 439
clear security flow session services-offload family on page 439
clear security flow session services-offload interface on page 439
clear security flow session services-offload logical-system on page 439
clear security flow session services-offload protocol on page 440
clear security flow session services-offload root-logical-system on page 440
clear security flow session services-offload source-port on page 440
clear security flow session services-offload source-prefix on page 440

Output Fields

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

Sample Output

clear security flow session services-offload

user@host>**clear security flow session services-offload**

```
0 active sessions cleared
```

clear security flow session services-offload application

user@host>**clear security flow session services-offload dns**

```
0 active sessions cleared
```

clear security flow session services-offload destination-port

user@host>**clear security flow session services-offload destination-port 1**

```
0 active sessions cleared
```

clear security flow session services-offload destination-prefix

user@host>**clear security flow session services-offload destination-prefix 10.0.0.1**

```
0 active sessions cleared
```

clear security flow session services-offload family

user@host>**clear security flow session services-offload family inet**

```
1 active sessions cleared
```

clear security flow session services-offload interface

user@host>**clear security flow session services-offload interface ge-0/0/0.0**

```
0 active sessions cleared
```

clear security flow session services-offload logical-system

user@host>**clear security flow session services-offload logical-system all**

```
0 active sessions cleared
```

clear security flow session services-offload protocol

user@host>**clear security flow session services-offload protocol pim**

```
0 active sessions cleared
```

clear security flow session services-offload root-logical-system

user@host>**clear security flow session services-offload root-logical-system application dns**

```
0 active sessions cleared
```

clear security flow session services-offload source-port

user@host>**clear security flow session services-offload source-port 1**

```
0 active sessions cleared
```

clear security flow session services-offload source-prefix

user@host>**clear security flow session services-offload source-prefix 10.0.0.1**

```
0 active sessions cleared
```

clear security flow session session-identifier

Syntax

```
clear security flow session session-identifier
session-identifier
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5; **node** options added in Junos OS Release 9.0.

Description

Clear the session with the specific identifier.

Options

- ***session-identifier*** —Number from 1 through 4,294,967,295 that identifies the security session.
- **node**—(Optional) For chassis cluster configurations, clear the specified session on a specific node (device) in the cluster.
  - ***node-id***—Identification number of the node. It can be 0 or 1.
  - **all**—Clear all nodes.
  - **local**—Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session session-identifier | 682

List of Sample Output

clear security flow session session-identifier 1 on page 442
clear security flow session session-identifier 1 node 0 on page 442

Output Fields

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

Sample Output

clear security flow session session-identifier 1

```
user@host> clear security flow session session-identifier 1

0 active sessions cleared
```

Sample Output

clear security flow session session-identifier 1 node 0

```
user@host> clear security flow session session-identifier 1 node 0

node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session source-port

Syntax

```
clear security flow session source-port
source-port-number
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5; **node** options added in Junos OS Release 9.0.

Description

Clear each session that uses the specified source port.

Options

- ***source-port-number*** —Number that identifies the source port.

  Range: 1 through 65,535

- **node**—(Optional) For chassis cluster configurations, clear sessions on the specified source port on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Clear all nodes.
  - **local** —Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session source-port | 688

List of Sample Output

clear security flow session source-port 1 on page 444
clear security flow session source-port 1 node 0 on page 444

Output Fields

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

Sample Output

clear security flow session source-port 1

user@host> **clear security flow session source-port 1**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

Sample Output

clear security flow session source-port 1 node 0

user@host> **clear security flow session source-port 1 node 0**

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session source-prefix

Syntax

```
clear security flow session source-prefix
source-prefix-number
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Release 8.5 of Junos OS.
The **node** options added in Release 9.0 of Junos OS.
Support for IPv6 addresses added in Release 10.2 of Junos OS.

Description

Clear sessions that match the source prefix.

Options

- ***source-prefix-number***—Source IP prefix or address.
- **node**—(Optional) For chassis cluster configurations, clear security sessions matching the source prefix on a specific node (device) in the cluster.
  - ***node-id***—Identification number of the node. It can be 0 or 1.
  - **all**—Clear all nodes.
  - **local**—Clear the local node.
  - **primary**—Clear the primary node.

Required Privilege Level

clear

RELATED DOCUMENTATION

show security flow session source-prefix | 693

List of Sample Output

clear security flow session source-prefix 10.0.0.1 on page 446
clear security flow session source-prefix 10::10 on page 446
clear security flow session source-prefix 10.0.0.1 node 0 on page 446

Output Fields

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

Sample Output

clear security flow session source-prefix 10.0.0.1

```
user@host> clear security flow session source-prefix 10.0.0.1
```

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
node1:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session source-prefix 10::10

```
user@host> clear security flow session source-prefix 10::10
```

```
1 active sessions cleared
```

Sample Output

clear security flow session source-prefix 10.0.0.1 node 0

```
user@host> clear security flow session source-prefix 10.0.0.1 node 0
```

```
node0:
--------------------------------------------------------------------------
0 active sessions cleared
```

clear security flow session tunnel

Syntax

```
clear security flow session tunnel [Enter] | tunnel-session-parameters
```

Release Information

Command introduced in Junos OS Release 8.5.

Description

Clear all active tunnel sessions by entering the command without parameters, or clear the tunnel session whose session parameters are specified.

Options

application—Name of the application or application set.

- **dns**—Domain Name System
- **ftp**—File Transfer Protocol
- **ignore**—Ignore application type
- **mgcp-ca**—Media Gateway Control Protocol with Call Agent
- **mgcp-ua**—MGCP with User Agent
- **ms-rpc**—Microsoft RPC
- **pptp**—Point-to-Point Tunneling Protocol
- **q931**—ISDN connection control protocol
- **ras**—RAS
- **realaudio**—RealAudio
- **rsh**—UNIX remote shell services
- **rtsp**—Real-Time Streaming Protocol
- **sccp**—Skinny Client Control Protocol
- **sip**—Session Initiation Protocol
- **sqlnet-v2**—Oracle SQLNET
- **sun-rpc**—Sun Microsystems RPC
- **talk**—TALK program
- **tftp**—Trivial File Transfer Protocol

To display the supported applications list on an SRX Series device, enter the following command from configuration mode:

```
user@host# show groups junos-default applications
```

conn-tag—session connection identifiers are in the following range:

Range: 0 through 4294967295

destination-port—Destination port.

Range: 1 through 65535

destination-prefix—Destination IP prefix or address.

family— Protocol family:

- inet—Clear IPv4 sessions
- inet6–Clear IPv6 sessions

interface—Name of incoming or outgoing interface.

protocol —IP protocol number.

source-port— Source port.

Range: 1 through 65535.

source-prefix—Source IP prefix or address.

Required Privilege Level
clear

RELATED DOCUMENTATION

show security flow session tunnel

Output Fields

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

The same output is displayed when you enter an option for this command.

Sample Output

`user@host>` **clear security flow session tunnel**

```
number active sessions cleared
```

clear security forward-options mirror filter

Syntax

```
clear security forward-options mirror-filter (all | filter-name)
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10.

Description

Clear statistics about configured mirror filters.

Options

all—Clear statistics for all configured mirror filters.

filter-name—Clear statistics for the specified mirror filter.

Required Privilege Level

clear

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346

show security forwarding-options mirror-filter | 736

monitor security flow file

Syntax

```
monitor security flow file
<file-name>
<files number>
<match regular-expression>
<size maximum-file-size>
<(world-readable | no-world-readable)>
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10.

Description

Configure options for the security flow monitoring output.

Options

filename—Name of the file to receive the output of the monitoring operation. All output is saved in the **/var/log/** directory.

files *number*—Maximum number of output files. If you specify a maximum number of files, you must also specify a maximum file size with the **size** option.

Range: 2 through 1000 files

Default: 10 files

match *regular-expression*—Refine the output to include lines that contain the regular expression.

size *maximum-file-size*—Maximum size of each output file. When an output file named ***output*** reaches this size, it is renamed ***output*.0**. When the output file again reaches its maximum size, ***output*.0** is renamed ***output*.1** and **output** file is renamed ***output*.0**. This renaming scheme continues until the maximum number of output files is reached. Then the oldest output file is overwritten.

If you specify a maximum file size, you also must specify a maximum number of output files with the **files** option.

Range: 10 KB through 1 GB

Default: 128 KB

(world-readable | no-world-readable)—By default, the output files can be accessed only by the user who configures the monitoring operation. The **world-readable** option enables all users to read the file. To explicitly set the default behavior, use the **no-world-readable** option.

Required Privilege Level

trace

RELATED DOCUMENTATION

Monitoring Security Flow Sessions Overview | 186

monitor security flow filter | 453

monitor security flow start | 455

show monitor security flow | 540

Output Fields

This command produces no output.

monitor security flow filter

Syntax

```
monitor security flow filter filter-name
<conn-tag session-connection-tag>
<destination-port (port-range | protocol-name)>
<destination-prefix destination-prefix>
<interface interface-name>
<logical-system logical-system-name>
<protocol (protocol name | protocol number)>
<root-logical-system>
<source-port (port-range | protocol-name)>
<source-prefix source-prefix>
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10. The was updated in Junos OS Release 15.1X49-D70 with the addition of the conn-tag filter parameter.

Description

Set security flow filters to define flow sessions that you want to monitor. A maximum of 64 filters is supported at a time.

Defining the filters themselves does not trigger monitoring. You must explicitly use the **monitor security flow start** command to enable monitoring. Once monitoring starts, any traffic that matches the specified filters is saved in an output file in the **/var/log/** directory.

> **NOTE:** Unlike filters defined in the configuration mode, these filters defined using operational mode commands are cleared when you reboot your system. They are used expressly for debugging purposes.

Options

filter *filter-name*—Specify a name for the filter. The filter name can contain letters, numbers, underscores (_) and hyphens (-) and can be up to 64 characters long.

conn-tag—Specify the session connection tag. The session connection tag uniquely identifies a session.

destination-port (*port-range* | *protocol-name*)—Specify the TCP or UDP destination port to match. You can also specify a range of TCP or UDP destination ports and monitor all traffic in this group.

destination-prefix *destination-prefix*—Specify the destination IPv4 or IPv6 address prefix to match.

interface *interface-name*—Specify the logical interface name to match.

logical-system *logical-system-name*—Specify the logical system name to match.

protocol (*protocol name* | *protocol number*)—Specify the IP protocol type to match.

root-logical-system—(Default) Specify the root logical system to match.

source-port (*port-range* | *protocol-name*)—Specify the TCP or UDP source port to match. You can also specify a range of TCP or UDP source ports and monitor all traffic in this group.

source-prefix *source-prefix*—Specify the source IP address prefix to match.

Required Privilege Level
view

RELATED DOCUMENTATION

- Monitoring Security Flow Sessions Overview | **186**
- monitor security flow file | **451**
- monitor security flow start | **455**
- monitor security flow stop | **456**

monitor security flow start

Syntax

```
monitor security flow start
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10.

Description

Start the monitoring of security flow session. Once monitoring starts, any traffic that matches the specified filters is saved in an output file in the **var/log/** directory. At least one filter must be defined for the monitoring to start.

Use the **monitor security flow stop** command to stop the monitoring of flow sessions.

Options

This command has no options.

Required Privilege Level

trace

RELATED DOCUMENTATION

- Monitoring Security Flow Sessions Overview | 186
- show monitor security flow | 540
- monitor security flow filter | 453
- monitor security flow stop | 456

Output Fields

This command produces no output.

monitor security flow stop

Syntax

```
monitor security flow stop
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10.

Description

Stop monitoring the security flow session. Use the **monitor security flow start** command to start the monitoring of flow sessions.

Options

This command has no options.

Required Privilege Level

trace

RELATED DOCUMENTATION

Monitoring Security Flow Sessions Overview | 186

monitor security flow start | 455

Output Fields

This command produces no output.

show chassis environment (Security)

Syntax

```
show chassis environment
```

Release Information
Command introduced in Junos OS Release 9.2.

Description
Display environmental information about the services gateway chassis, including the temperature and information about the fans, power supplies, and Routing Engine.

Options
none—Display environmental information about the device.

cb *slot-number*—Display chassis environmental information for the Control Board.

fpc *fpc-slot*—Display chassis environmental information for a specified Flexible PIC Concentrator.

fpm—Display chassis environmental information for the craft interface (FPM).

node—Display node specific chassis information.

pem *slot-number*—Display chassis environmental information for the specified Power Entry Module.

routing-engine *slot-number*—Display chassis environmental information for the specified Routing Engine.

Required Privilege Level
view

RELATED DOCUMENTATION

show chassis hardware (View) | **478**

List of Sample Output
show chassis environment on page 458
show chassis environment fpc (SRX5800, SRX5400, and SRX5600) on page 462

Output Fields
Table 25 on page 458 lists the output fields for the **show chassis environment** command. Output fields are listed in the approximate order in which they appear.

Table 25: show chassis environment Output Fields

Field Name	Field Description
Temp	Temperature of air flowing through the chassis in degrees Celsius (C) and Fahrenheit (F).
Fan	Fan status: **OK**, **Testing** (during initial power-on), **Failed**, or **Absent**.

Sample Output

show chassis environment

`user@host>` **show chassis environment**

```
user@host> show chassis environment
Class Item                          Status     Measurement
Temp  PEM 0                         OK         40 degrees C / 104 degrees F
      PEM 1                         OK         40 degrees C / 104 degrees F
      PEM 2                         OK         40 degrees C / 104 degrees F
      PEM 3                         OK         45 degrees C / 113 degrees F
      Routing Engine 0              OK         31 degrees C / 87 degrees F
      Routing Engine 0 CPU          OK         27 degrees C / 80 degrees F
      Routing Engine 1              Absent
      Routing Engine 1 CPU          Absent
      CB 0 Intake                   OK         28 degrees C / 82 degrees F
      CB 0 Exhaust A                OK         27 degrees C / 80 degrees F
      CB 0 Exhaust B                OK         29 degrees C / 84 degrees F
      CB 0 ACBC                     OK         29 degrees C / 84 degrees F
      CB 0 SF A                     OK         36 degrees C / 96 degrees F
      CB 0 SF B                     OK         31 degrees C / 87 degrees F
      CB 1 Intake                   OK         27 degrees C / 80 degrees F
      CB 1 Exhaust A                OK         26 degrees C / 78 degrees F
      CB 1 Exhaust B                OK         29 degrees C / 84 degrees F
      CB 1 ACBC                     OK         27 degrees C / 80 degrees F
      CB 1 SF A                     OK         36 degrees C / 96 degrees F
      CB 1 SF B                     OK         31 degrees C / 87 degrees F
      CB 2 Intake                   Absent
      CB 2 Exhaust A                Absent
      CB 2 Exhaust B                Absent
      CB 2 ACBC                     Absent
      CB 2 XF A                     Absent
      CB 2 XF B                     Absent
      FPC 0 Intake                  OK         47 degrees C / 116 degrees F
```

```
FPC 0 Exhaust A                 OK          44 degrees C / 111 degrees F
FPC 0 Exhaust B                 OK          52 degrees C / 125 degrees F
FPC 0 xlp0 TSen                 OK          51 degrees C / 123 degrees F
FPC 0 xlp0 Chip                 OK          46 degrees C / 114 degrees F
FPC 0 xlp1 TSen                 OK          51 degrees C / 123 degrees F
FPC 0 xlp1 Chip                 OK          47 degrees C / 116 degrees F
FPC 0 xlp2 TSen                 OK          44 degrees C / 111 degrees F
FPC 0 xlp2 Chip                 OK          42 degrees C / 107 degrees F
FPC 0 xlp3 TSen                 OK          48 degrees C / 118 degrees F
FPC 0 xlp3 Chip                 OK          43 degrees C / 109 degrees F
FPC 1 Intake                    OK          41 degrees C / 105 degrees F
FPC 1 Exhaust A                 OK          41 degrees C / 105 degrees F
FPC 1 Exhaust B                 OK          51 degrees C / 123 degrees F
FPC 1 LU TSen                   OK          46 degrees C / 114 degrees F
FPC 1 LU Chip                   OK          45 degrees C / 113 degrees F
FPC 1 XM TSen                   OK          46 degrees C / 114 degrees F
FPC 1 XM Chip                   OK          52 degrees C / 125 degrees F
FPC 1 xlp0 TSen                 OK          49 degrees C / 120 degrees F
FPC 1 xlp0 Chip                 OK          42 degrees C / 107 degrees F
FPC 1 xlp1 TSen                 OK          49 degrees C / 120 degrees F
FPC 1 xlp1 Chip                 OK          44 degrees C / 111 degrees F
FPC 1 xlp2 TSen                 OK          38 degrees C / 100 degrees F
FPC 1 xlp2 Chip                 OK          39 degrees C / 102 degrees F
FPC 1 xlp3 TSen                 OK          44 degrees C / 111 degrees F
FPC 1 xlp3 Chip                 OK          42 degrees C / 107 degrees F
FPC 2 Intake                    OK          29 degrees C / 84 degrees F
FPC 2 Exhaust A                 OK          34 degrees C / 93 degrees F
FPC 2 Exhaust B                 OK          40 degrees C / 104 degrees F
FPC 2 I3 0 TSensor              OK          42 degrees C / 107 degrees F
FPC 2 I3 0 Chip                 OK          41 degrees C / 105 degrees F
FPC 2 I3 1 TSensor              OK          40 degrees C / 104 degrees F
FPC 2 I3 1 Chip                 OK          39 degrees C / 102 degrees F
FPC 2 I3 2 TSensor              OK          38 degrees C / 100 degrees F
FPC 2 I3 2 Chip                 OK          37 degrees C / 98 degrees F
FPC 2 I3 3 TSensor              OK          35 degrees C / 95 degrees F
FPC 2 I3 3 Chip                 OK          35 degrees C / 95 degrees F
FPC 2 IA 0 TSensor              OK          45 degrees C / 113 degrees F
FPC 2 IA 0 Chip                 OK          42 degrees C / 107 degrees F
FPC 2 IA 1 TSensor              OK          41 degrees C / 105 degrees F
FPC 2 IA 1 Chip                 OK          43 degrees C / 109 degrees F
FPC 9 Intake                    OK          29 degrees C / 84 degrees F
FPC 9 Exhaust A                 OK          41 degrees C / 105 degrees F
FPC 9 Exhaust B                 OK          48 degrees C / 118 degrees F
FPC 9 LU TSen                   OK          48 degrees C / 118 degrees F
```

```
FPC 9 LU Chip                OK          47 degrees C / 116 degrees F
FPC 9 XM TSen                OK          48 degrees C / 118 degrees F
FPC 9 XM Chip                OK          54 degrees C / 129 degrees F
FPC 9 xlp0 TSen              OK          45 degrees C / 113 degrees F
FPC 9 xlp0 Chip              OK          42 degrees C / 107 degrees F
FPC 9 xlp1 TSen              OK          49 degrees C / 120 degrees F
FPC 9 xlp1 Chip              OK          46 degrees C / 114 degrees F
FPC 9 xlp2 TSen              OK          37 degrees C / 98 degrees F
FPC 9 xlp2 Chip              OK          40 degrees C / 104 degrees F
FPC 9 xlp3 TSen              OK          45 degrees C / 113 degrees F
FPC 9 xlp3 Chip              OK          41 degrees C / 105 degrees F
FPC 10 Intake                OK          32 degrees C / 89 degrees F
FPC 10 Exhaust A             OK          44 degrees C / 111 degrees F
FPC 10 Exhaust B             OK          53 degrees C / 127 degrees F
FPC 10 LU 0 TSen             OK          43 degrees C / 109 degrees F
FPC 10 LU 0 Chip             OK          52 degrees C / 125 degrees F
FPC 10 LU 1 TSen             OK          43 degrees C / 109 degrees F
FPC 10 LU 1 Chip             OK          44 degrees C / 111 degrees F
FPC 10 LU 2 TSen             OK          43 degrees C / 109 degrees F
FPC 10 LU 2 Chip             OK          50 degrees C / 122 degrees F
FPC 10 LU 3 TSen             OK          43 degrees C / 109 degrees F
FPC 10 LU 3 Chip             OK          58 degrees C / 136 degrees F
FPC 10 XM 0 TSen             OK          43 degrees C / 109 degrees F
FPC 10 XM 0 Chip             OK          53 degrees C / 127 degrees F
FPC 10 XF 0 TSen             OK          43 degrees C / 109 degrees F
FPC 10 XF 0 Chip             OK          64 degrees C / 147 degrees F
FPC 10 PLX Switch TSen       OK          43 degrees C / 109 degrees F
FPC 10 PLX Switch Chip       OK          44 degrees C / 111 degrees F
FPC 11 Intake                OK          32 degrees C / 89 degrees F
FPC 11 Exhaust A             OK          41 degrees C / 105 degrees F
FPC 11 Exhaust B             OK          56 degrees C / 132 degrees F
FPC 11 LU 0 TSen             OK          45 degrees C / 113 degrees F
FPC 11 LU 0 Chip             OK          50 degrees C / 122 degrees F
FPC 11 LU 1 TSen             OK          45 degrees C / 113 degrees F
FPC 11 LU 1 Chip             OK          47 degrees C / 116 degrees F
FPC 11 LU 2 TSen             OK          45 degrees C / 113 degrees F
FPC 11 LU 2 Chip             OK          52 degrees C / 125 degrees F
FPC 11 LU 3 TSen             OK          45 degrees C / 113 degrees F
FPC 11 LU 3 Chip             OK          60 degrees C / 140 degrees F
FPC 11 XM 0 TSen             OK          45 degrees C / 113 degrees F
FPC 11 XM 0 Chip             OK          56 degrees C / 132 degrees F
FPC 11 XF 0 TSen             OK          45 degrees C / 113 degrees F
FPC 11 XF 0 Chip             OK          65 degrees C / 149 degrees F
FPC 11 PLX Switch TSen       OK          45 degrees C / 113 degrees F
```

```
      FPC 11 PLX Switch Chip         OK         46 degrees C / 114 degrees F
Fans  Top Fan Tray Temp              OK         34 degrees C / 93 degrees F
      Top Tray Fan 1                 OK         Spinning at normal speed
      Top Tray Fan 2                 OK         Spinning at normal speed
      Top Tray Fan 3                 OK         Spinning at normal speed
      Top Tray Fan 4                 OK         Spinning at normal speed
      Top Tray Fan 5                 OK         Spinning at normal speed
      Top Tray Fan 6                 OK         Spinning at normal speed
      Top Tray Fan 7                 OK         Spinning at normal speed
      Top Tray Fan 8                 OK         Spinning at normal speed
      Top Tray Fan 9                 OK         Spinning at normal speed
      Top Tray Fan 10                OK         Spinning at normal speed
      Top Tray Fan 11                OK         Spinning at normal speed
      Top Tray Fan 12                OK         Spinning at normal speed
      Bottom Fan Tray Temp           OK         31 degrees C / 87 degrees F
      Bottom Tray Fan 1              OK         Spinning at normal speed
      Bottom Tray Fan 2              OK         Spinning at normal speed
      Bottom Tray Fan 3              OK         Spinning at normal speed
      Bottom Tray Fan 4              OK         Spinning at normal speed
      Bottom Tray Fan 5              OK         Spinning at normal speed
      Bottom Tray Fan 6              OK         Spinning at normal speed
      Bottom Tray Fan 7              OK         Spinning at normal speed
      Bottom Tray Fan 8              OK         Spinning at normal speed
      Bottom Tray Fan 9              OK         Spinning at normal speed
      Bottom Tray Fan 10             OK         Spinning at normal speed
      Bottom Tray Fan 11             OK         Spinning at normal speed
      Bottom Tray Fan 12             OK         Spinning at normal speed
       OK
```

When you enter the **show chassis environment pem** command, the sample output is shown for DC PEM.

`user@host>` **show chassis environment pem**

```
node0:
--------------------------------------------------------------------------
PEM 0 status:
  State                      Online
  Temperature                OK
  DC Input:                  OK
  DC Output          Voltage(V) Current(A)  Power(W)  Load(%)
                         50        12            600      24
PEM 1 status:
  State                      Online
  Temperature                OK
```

```
  DC Input:                  OK
  DC Output           Voltage(V) Current(A)  Power(W)  Load(%)
                          50        31            1550     63

node1:
--------------------------------------------------------------------------
PEM 0 status:
  State                      Online
  Temperature                OK
  DC Input:                  OK
  DC Output           Voltage(V) Current(A)  Power(W)  Load(%)
                          50        12            600      24
PEM 1 status:
  State                      Online
  Temperature                OK
  DC Input:                  OK
  DC Output           Voltage(V) Current(A)  Power(W)  Load(%)
                          49        31            1519     62
```

show chassis environment fpc (SRX5800, SRX5400, and SRX5600)

`user@host>` **show chassis environment fpc**

```
FPC 1 status:
  State                      Online
  Temperature Intake         34 degrees C / 93 degrees F
  Temperature Exhaust A      48 degrees C / 118 degrees F
  Temperature Exhaust B      48 degrees C / 118 degrees F
  Temperature CPU0 DTS       55 degrees C / 131 degrees F
  Temperature CPU1 DTS       60 degrees C / 140 degrees F
  Temperature CPU2 DTS       54 degrees C / 129 degrees F
  Temperature CPU3 DTS       70 degrees C / 158 degrees F
  Temperature Talus 0        106 degrees C / 222 degrees F
  Temperature Middle 0       40 degrees C / 104 degrees F
  Temperature Talus 1        76 degrees C / 168 degrees F
  Temperature Middle 1       67 degrees C / 152 degrees F
  Power
         TALUS0-1.20V               1199 mV   14187 mA   17010 mW
         TALUS0-0.90V                900 mV    5000 mA    4500 mW
         BIAS0-3.30V                3299 mV    3769 mA   12433 mW
         PIC0_CPU_memory_CD-1.20    1199 mV    3781 mA    4533 mW
         USB0-5.00V                 5000 mV     155 mA     775 mW
         PIC0_CPU_memory_AB-1.20    1200 mV    5820 mA    6984 mW
```

```
PCH0-1.05V                  1050 mV    3582 mA    3761 mW
TALUS1-1.20V                1199 mV   13640 mA   16354 mW
TALUS1-0.90V                 899 mV    4679 mA    4206 mW
BIAS1-3.30V                 3300 mV    3175 mA   10477 mW
PIC1_CPU_memory_GH-1.20     1200 mV    4648 mA    5577 mW
USB1-5.00V                  4999 mV     346 mA    1729 mW
PIC1_CPU_memory_EF-1.20     1200 mV    5218 mA    6261 mW
PCH1-1.05V                  1050 mV    3328 mA    3494 mW
TPS53641-CPU0               1750 mV   46062 mA   80608 mW
TPS53641-CPU1               1739 mV   47437 mA   82492 mW
TPS53641-CPU2               1750 mV   45250 mA   79187 mW
TPS53641-CPU3               1739 mV   46875 mA   81515 mW
ETH-1.00V                    994 mV    2674 mA    2657 mW
TALUS0_Core-0.85V            849 mV   37750 mA   32049 mW
TALUS1_Core-0.85V            849 mV   26750 mA   22710 mW
Power_Brick1-12.00V        12001 mV   21000 mA  252021 mW
Power_Brick2-12.00V        11998 mV   23125 mA  277453 mW
PIM4820_48V0-48.00V        58392 mV   10286 mA  600620 mW
I2C Slave Revision         0
```

show chassis fpc (View)

Syntax

```
show chassis fpc
<detail < fpc-slot >| <node ( node-id  | local | primary)>> |
<node ( node-id  | local | primary)> |
<pic-status < fpc-slot >| <node ( node-id  | local | primary)>>
```

Release Information

Command modified in Junos OS Release 9.2.

Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the SRX5K-MPC3-100G10G (IOC3) and the SRX5K-MPC3-40G10G (IOC3) are introduced.

> **NOTE:** On SRX5K-MPC3-40G10G (IOC3), all four PICs cannot be powered on. A maximum of two PICs can be powered on at the same time. By default, PIC0 and PIC1 are online.

Use the **set chassis fpc <slot> pic <pic> power off** command to choose the PICs you want to power on. When you use the **set chassis fpc <slot> pic <pic> power off** command to power off PIC0 and PIC1, PIC2 and PIC3 are automatically turned on.

When you switch from one set of PICs to another set of PICs using the **set chassis fpc <slot> pic <pic> power off** command again, ensure that there is 60 seconds duration between the two actions, otherwise core files are seen during the configuration.

The Table 26 on page 464 summarizes the SRX5K-MPC3-40G10G (IOC3) PICs selected for various configuration scenarios.

Table 26: SRX5K-MPC3-40G10G (IOC3) PIC Selection Summary

CLI Configuration	PIC Selection
Default (i.e. no CLI configuration)	Online: PIC-0, PIC-1 Offline: PIC-2, PIC-3
PIC-1, PIC-2 and PIC-3 powered OFF	Online: PIC-0 Offline: PIC-1, PIC-2, PIC-3
PIC-0, PIC-2 and PIC-3 powered OFF	Online: PIC-1 Offline: PIC-0, PIC-2, PIC-3

Table 26: SRX5K-MPC3-40G10G (IOC3) PIC Selection Summary *(continued)*

CLI Configuration	PIC Selection
PIC-0, PIC-1 and PIC-3 powered OFF	Online: PIC-2 Offline: PIC-0, PIC-1, PIC-3
PIC-0, PIC-1 and PIC-2 powered OFF	Online: PIC-3 Offline: PIC-0, PIC-1, PIC-2
PIC-2 and PIC-3 powered OFF	Online: PIC-0, PIC-1 Offline: PIC-2, PIC-3
PIC-2 and PIC-3 powered OFF	Online: PIC-0, PIC-1 Offline: PIC-2, PIC-3
PIC-1 and PIC-2 powered OFF	Online: PIC-0, PIC-3 Offline: PIC-1, PIC-2
PIC-0 and PIC-3 powered OFF	Online: PIC-2, PIC-1 Offline: PIC-0, PIC-3
PIC-0 and PIC-1 powered OFF	Online: PIC-2, PIC-3 Offline: PIC-0, PIC-1
All other combinations of PICs being powered OFF (Invalid)	Online: PIC-0, PIC-1 Offline: PIC-2, PIC-3 Default PICs will be selected for the invalid combinations. Also, a system log message will be displayed to indicate the invalid combination PIC selection.

Description

Display status information about the installed Flexible PIC Concentrators (FPCs) and PICs.

Options

- none—Display status information for all FPCs.
- **detail**—(Optional) Display detailed FPC status information.

- ***fpc-slot*** —(Optional) Display information about the FPC in this slot.
- **node**—(Optional) For chassis cluster configurations, display status information for all FPCs or for the specified FPC on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.
- **pic-status**—(Optional) Display status information for all FPCs or for the FPC in the specified slot (see ***fpc-slot***).

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Interfaces

List of Sample Output
show chassis fpc on page 468
show chassis fpc (SRX5600 and SRX5800 devices) on page 468
show chassis fpc (SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G (IOC3) or SRX5K-MPC3-40G10G (IOC3) on page 468
show chassis fpc detail 2 on page 469
show chassis fpc pic-status (SRX5600 and SRX5800 devices) on page 469
show chassis fpc pic-status (SRX5600 and SRX5800 devices with SPC2) on page 470
show chassis fpc pic-status (SRX5600 and SRX5800 devices with SRX5K-MPC) on page 470
show chassis fpc pic-status (SRX5600 and SRX5800 devices when Express Path [formerly known as services offloading] is configured) on page 471
show chassis fpc pic-status (with 20-Gigabit Ethernet MIC with SFP) on page 471
show chassis fpc pic-status(SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G (IOC3) or SRX5K-MPC3-40G10G (IOC3 and when Express Path [formerly known as services offloading] is configured) on page 472
show chassis fpc pic-status for HA (SRX5600 and SRX5800 devices) on page 473
show chassis fpc pic-status for HA(SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G (IOC3) or SRX5K-MPC3-40G10G (IOC3) on page 473

Output Fields

Table 27 on page 467 lists the output fields for the **show chassis fpc** command. Output fields are listed in the approximate order in which they appear.

Table 27: show chassis fpc Output Fields

Field Name	Field Description
Slot or Slot State	Slot number and state. The state can be one of the following conditions: • **Dead**—Held in reset because of errors. • **Diag**—Slot is being ignored while the device is running diagnostics. • **Dormant**—Held in reset. • **Empty**—No FPC is present. • **Online**—FPC is online and running. • **Present**—FPC is detected by the device, but is either not supported by the current version of Junos OS or inserted in the wrong slot. The output also states either **Hardware Not Supported** or **Hardware Not In Right Slot**. FPC is coming up but not yet online. • **Probed**—Probe is complete; awaiting restart of the Packet Forwarding Engine (PFE). • **Probe-wait**—Waiting to be probed.
Temp (C) or Temperature	Temperature of the air passing by the FPC, in degrees Celsius or in both Celsius and Fahrenheit.
Total CPU Utilization (%)	Total percentage of CPU being used by the FPC's processor.
Interrupt CPU Utilization (%)	Of the total CPU being used by the FPC's processor, the percentage being used for interrupts.
Memory DRAM (MB)	Total DRAM, in megabytes, available to the FPC's processor.
Heap Utilization (%)	Percentage of heap space (dynamic memory) being used by the FPC's processor. If this number exceeds 80 percent, there may be a software problem (memory leak).
Buffer Utilization (%)	Percentage of buffer space being used by the FPC's processor for buffering internal messages.
Start Time	Time when the Routing Engine detected that the FPC was running.
Uptime	How long the Routing Engine has been connected to the FPC and, therefore, how long the FPC has been up and running.
PIC type	(**pic-status** output only) Type of FPC.

Sample Output

show chassis fpc

```
user@host> show chassis fpc
```

```
                     Temp   CPU Utilization (%)     Memory    Utilization (%)
     Slot State        (C)    Total  Interrupt        DRAM (MB) Heap  Buffer
      0  Online          --------------------- CPU less FPC --------------------
      1  Online          ---------------------  Not Usable --------------------
      2  Online          --------------------- CPU less FPC --------------------
```

show chassis fpc (SRX5600 and SRX5800 devices)

```
user@host> show chassis fpc
```

```
                     Temp   CPU Utilization (%)       Memory    Utilization (%)
Slot State           (C)   Total  Interrupt         DRAM (MB) Heap     Buffer
  0  Empty
  1  Empty
  2  Empty
  3  Online           37       3          0             1024        7     42
  4  Empty
  5  Empty
  6  Online           30        8         0            1024        23    30
  7  Empty
  8  Empty
  9  Empty
 10  Empty
 11  Empty
```

show chassis fpc
(SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G (IOC3) or SRX5K-MPC3-40G10G (IOC3)

```
user@host> show chassis fpc
```

```
                      Temp  CPU Utilization (%)   CPU Utilization (%)  Memory

                                  Utilization (%)
Slot State           (C)  Total  Interrupt      1min   5min   15min  DRAM (MB)

                                Heap     Buffer
  0  Online           36     20          0       20     19     19    1024
```

```
                                    4         26
  1  Online            35      8          0        8      8      8    2048

                                   12         14
  2  Online            40     21          0       20     20     20    3584

                                    5         13
```

Sample Output

show chassis fpc detail 2

```
user@host> show chassis fpc detail 2
```

```
Slot 2 information:
  State                                 Online
  Temperature                            37
  Total CPU DRAM                       1024 MB
  Total RLDRAM                          0 MB
  Total DDR DRAM                        0 MB
  Start time:                             2012-07-18 07:18:50 PDT
  Uptime:                                 4 days, 21 hours, 51 minutes, 59 seconds
  Max Power Consumption                 0 Watts
```

Sample Output

show chassis fpc pic-status (SRX5600 and SRX5800 devices)

```
user@host> show chassis fpc pic-status
```

```
Slot 3   Online       SRX5k SPC
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
Slot 6   Online       SRX5k DPC 4x 10GE
  PIC 0  Online       1x 10GE(LAN/WAN) RichQ
  PIC 1  Online       1x 10GE(LAN/WAN) RichQ
  PIC 2  Online       1x 10GE(LAN/WAN) RichQ
  PIC 3  Online       1x 10GE(LAN/WAN) RichQ
```

show chassis fpc pic-status (SRX5600 and SRX5800 devices with SPC2)

user@host> **show chassis fpc pic-status**

```
Slot 0   Online       SRX5k DPC 40x 1GE
  PIC 0  Online       10x 1GE RichQ
  PIC 1  Online       10x 1GE RichQ
  PIC 2  Online       10x 1GE RichQ
  PIC 3  Online       10x 1GE RichQ
Slot 2   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 3   Online       SRX5k SPC II
  PIC 0  Online       SPU Flow
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 5   Online       SRX5k SPC
  PIC 0  Online       SPU Flow
  PIC 1  Online       SPU Flow
```

show chassis fpc pic-status (SRX5600 and SRX5800 devices with SRX5K-MPC)

user@host> **show chassis fpc pic-status**

```
Slot 0   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 1   Online       SRX5k SPC II
  PIC 0  Online       SPU Flow
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 2   Online       SRX5k DPC 4X 10GE
  PIC 0  Online       1x 10GE(LAN/WAN) RichQ
  PIC 1  Online       1x 10GE(LAN/WAN) RichQ
  PIC 2  Online       1x 10GE(LAN/WAN) RichQ
  PIC 3  Online       1x 10GE(LAN/WAN) RichQ
Slot 6   Offline      SRX5k SPC II
```

```
Slot 9   Online       SRX5k SPC II
  PIC 0  Online       SPU Flow
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 10  Online       SRX5k IOC II
  PIC 0  Online       10x 10GE SFP+
  PIC 2  Online       1x 100GE CFP
Slot 11  Online       SRX5k IOC II
  PIC 0  Online       1x 100GE CFP
  PIC 2  Online       2x 40GE QSFP+
```

show chassis fpc pic-status (SRX5600 and SRX5800 devices when Express Path [formerly known as services offloading] is configured)

`user@host>` **show chassis fpc pic-status**

```
Slot 0   Offline      SRX5k DPC 40x 1GE
Slot 1   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 2   Offline      SRX5k SPC
Slot 4   Online       SRX5k IOC3 24XGE+6XLG
  PIC 2  Online       3x 40GE QSFP+- np-cache/services-offload
  PIC 3  Online       3x 40GE QSFP+- np-cache/services-offload
Slot 5   Online       SRX5k IOC II
  PIC 0  Online       10x 1GE(LAN) SFP- np-cache/services-offload
  PIC 1  Online       10x 1GE(LAN) SFP- np-cache/services-offload
  PIC 2  Online       10x 10GE SFP+- np-cache/services-offload
```

show chassis fpc pic-status (with 20-Gigabit Ethernet MIC with SFP)

`user@host>` **show chassis fpc pic-status**

```
node0:
--------------------------------------------------------------------------
Slot 0   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
```

```
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 1   Offline      SRX5k SPC II
Slot 2   Online       SRX5k DPC 4X 10GE
  PIC 0  Online       1x 10GE(LAN/WAN) RichQ
  PIC 1  Online       1x 10GE(LAN/WAN) RichQ
  PIC 2  Online       1x 10GE(LAN/WAN) RichQ
  PIC 3  Online       1x 10GE(LAN/WAN) RichQ
Slot 9   Online       SRX5k IOC II
  PIC 0  Online       10x 1GE(LAN) SFP
  PIC 1  Online       10x 1GE(LAN) SFP
  PIC 2  Online       10x 1GE(LAN) SFP
  PIC 3  Online       10x 1GE(LAN) SFP
Slot 10  Online       SRX5k IOC II
  PIC 0  Online       10x 10GE SFP+
  PIC 2  Online       1x 100GE CFP
Slot 11  Offline      SRX5k IOC II
```

show chassis fpc pic-status
(SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G (IOC3) or SRX5K-MPC3-40G10G (IOC3 and when Express Path [formerly known as services offloading] is configured)

```
user@host> show chassis fpc pic-status
```

```
Slot 0   Offline      SRX5k DPC 40x 1GE
Slot 1   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Online       SPU Flow
  PIC 3  Online       SPU Flow
Slot 2   Offline      SRX5k SPC
Slot 4   Online       SRX5k IOC3 24XGE+6XLG
  PIC 2  Online       3x 40GE QSFP+- np-cache/services-offload
  PIC 3  Online       3x 40GE QSFP+- np-cache/services-offload
Slot 5   Online       SRX5k IOC II
  PIC 0  Online       10x 1GE(LAN) SFP- np-cache/services-offload
  PIC 1  Online       10x 1GE(LAN) SFP- np-cache/services-offload
  PIC 2  Online       10x 10GE SFP+- np-cache/services-offload
```

Sample Output

show chassis fpc pic-status for HA (SRX5600 and SRX5800 devices)

user@host> **show chassis fpc pic-status**

```
node0:
--------------------------------------------------------------
Slot 4   Online       SRX5k DPC 40x 1GE
  PIC 0  Online       10x 1GE RichQ
  PIC 1  Online       10x 1GE RichQ
  PIC 2  Online       10x 1GE RichQ
  PIC 3  Online       10x 1GE RichQ
Slot 5   Online       SRX5k SPC
  PIC 0  Online       SPU Cp-Flow
  PIC 1  Online       SPU Flow

node1:
--------------------------------------------------------------
Slot 4   Online       SRX5k DPC 40x 1GE
  PIC 0  Online       10x 1GE RichQ
  PIC 1  Online       10x 1GE RichQ
  PIC 2  Online       10x 1GE RichQ
  PIC 3  Online       10x 1GE RichQ
Slot 5   Online       SRX5k SPC
  PIC 0  Online       SPU Cp-Flow
  PIC 1  Online       SPU Flow
```

show chassis fpc pic-status for HA (SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G (IOC3) or SRX5K-MPC3-40G10G (IOC3)

user@host> **show chassis fpc pic-status**

```
user@host> show chassis fpc pic-status
node0:
--------------------------------------------------------------------------
Slot 2   Online       SRX5k IOC3 24XGE+6XLG
  PIC 0  Online       12x 10GE SFP+
  PIC 1  Online       12x 10GE SFP+
  PIC 2  Offline      3x 40GE QSFP+
  PIC 3  Offline      3x 40GE QSFP+
Slot 4   Online       SRX5k IOC II
  PIC 2  Online       10x 10GE SFP+
Slot 5   Online       SRX5k SPC II
```

```
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Offline
  PIC 3  Offline

node1:
--------------------------------------------------------------------------
Slot 2   Online       SRX5k IOC3 24XGE+6XLG
  PIC 0  Online       12x 10GE SFP+
  PIC 1  Online       12x 10GE SFP+
  PIC 2  Offline      3x 40GE QSFP+
  PIC 3  Offline      3x 40GE QSFP+
Slot 4   Online       SRX5k IOC II
  PIC 2  Online       10x 10GE SFP+
Slot 5   Online       SRX5k SPC II
  PIC 0  Online       SPU Cp
  PIC 1  Online       SPU Flow
  PIC 2  Offline
  PIC 3  Offline
```

show chassis fpc errors

Syntax

```
show chassis fpc errors;
```

Release Information

Command introduced in Junos OS Release 15.1X49-D40.

Description

Display chassis error information including FPC number, severity of error, number of error occurred, cleared, threshold, and corresponding action.

Error Severity Level	Default Threshold	Default Action
Fatal	1	Get the current state and reset the FPC.
Major	1	Get the current state of the FPC and raise an alarm.
Minor	10	Write a log for the event.

Required Privilege Level

view

RELATED DOCUMENTATION

fpc error | 325

List of Sample Output

show chassis fpc errors on page 476

Output Fields

Table 28 on page 475 lists the output fields for the **show chassis fpc errors** command. Output fields are listed in the approximate order in which they appear.

Table 28: show chassis fpc errors Output Fields

Field Name	Field Description
FPC	The FPC number.

Table 28: show chassis fpc errors Output Fields *(continued)*

Field Name	Field Description
Level	The severity of the error. It can be configured as follows: • **fatal**—Fatal error on the FPC. • **major**—Major error on the FPC. • **minor**—Minor error on the FPC.
Occurred	Number of error instances that have occurred.
Cleared	Number of error instances that have been cleared.
Threshold	Configured threshold value. The associated detection and recovery actions are triggered when this threshold value is exceeded.
Action	The detection and recovery actions that are triggered when the threshold value is exceeded. • Restart the FPC. • Get the current state of the FPC and raise an alarm. • Write a log for the event.

Sample Output

show chassis fpc errors

`user@host>` **show chassis fpc errors**

```
FPC  Level Occurred Cleared Threshold Action-Taken Action
0   Minor      0      0     10          LOG|
    Major      0      0      1        GET STATE|ALARM|
    Fatal      0      0      1         RESET
    Pfe-State: pfe-0 -ENABLED | pfe-1 -ENABLED | pfe-2 -ENABLED | pfe-3 -ENABLED
| pfe-4 -ENABLED | pfe-5 -ENABLED | pfe-6 -ENABLED | pfe-7 -ENABLED |
1   Minor      0      0     10          LOG|
    Major      0      0      1        GET STATE|ALARM|
    Fatal      0      0      1         RESET
    Pfe-State: pfe-0 -ENABLED | pfe-1 -ENABLED | pfe-2 -ENABLED | pfe-3 -ENABLED
| pfe-4 -ENABLED | pfe-5 -ENABLED | pfe-6 -ENABLED | pfe-7 -ENABLED |
2   Minor      0      0     10          LOG|
    Major      0      0      1        GET STATE|ALARM|
    Fatal      0      0      1         RESET
    Pfe-State: pfe-0 -ENABLED | pfe-1 -ENABLED | pfe-2 -ENABLED | pfe-3 -ENABLED
```

```
| pfe-4 -ENABLED | pfe-5 -ENABLED | pfe-6 -ENABLED | pfe-7 -ENABLED |
4   Minor      0      0     10          LOG|
    Major      0      0      1        GET STATE|ALARM|
    Fatal      0      0      1         RESET
    Pfe-State: pfe-0 -ENABLED | pfe-1 -ENABLED | pfe-2 -ENABLED | pfe-3 -ENABLED
| pfe-4 -ENABLED | pfe-5 -ENABLED | pfe-6 -ENABLED | pfe-7 -ENABLED |
5   Minor      0      0     10          LOG|
    Major      0      0      1        GET STATE|ALARM|
    Fatal      0      0      1         RESET
    Pfe-State: pfe-0 -ENABLED | pfe-1 -ENABLED | pfe-2 -ENABLED | pfe-3 -ENABLED
| pfe-4 -ENABLED | pfe-5 -ENABLED | pfe-6 -ENABLED | pfe-7 -ENABLED |
6   Minor      0      0     10          LOG|
    Major      0      0      1        GET STATE|ALARM|
    Fatal      0      0      1         RESET
    Pfe-State: pfe-0 -ENABLED | pfe-1 -ENABLED | pfe-2 -ENABLED | pfe-3 -ENABLED
| pfe-4 -ENABLED | pfe-5 -ENABLED | pfe-6 -ENABLED | pfe-7 -ENABLED |
7   Minor      0      0     10          LOG|
    Major      0      0      1        GET STATE|ALARM|
    Fatal      0      0      1         RESET
    Pfe-State: pfe-0 -ENABLED | pfe-1 -ENABLED | pfe-2 -ENABLED | pfe-3 -ENABLED
| pfe-4 -ENABLED | pfe-5 -ENABLED | pfe-6 -ENABLED | pfe-7 -ENABLED |
```

show chassis hardware (View)

Syntax

```
show chassis hardware
<clei-models | detail | extensive | models | node ( node-id | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 9.2. Command modified in Junos OS Release 9.2 to include **node** option.

Description

Display chassis hardware information.

Options

- **clei-models**—(Optional) Display Common Language Equipment Identifier Code (CLEI) barcode and model number for orderable field-replaceable units (FRUs).
- **detail** | **extensive**—(Optional) Display the specified level of output.
- **models**—(Optional) Display model numbers and part numbers for orderable FRUs.
- **node**—(Optional) For chassis cluster configurations, display chassis hardware information on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

Interface Naming Conventions

Output Fields

Table 29 on page 479 lists the output fields for the **show chassis hardware** command. Output fields are listed in the approximate order in which they appear.

Table 29: show chassis hardware Output Fields

Field Name	Field Description
Item	Chassis component—Information about the backplane; power supplies; fan trays; Routing Engine; each Physical Interface Module (PIM)—reported as FPC and PIC—and each fan, blower, and impeller.
Version	Revision level of the chassis component.
Part Number	Part number for the chassis component.
Serial Number	Serial number of the chassis component. The serial number of the backplane is also the serial number of the device chassis. Use this serial number when you need to contact Juniper Networks Customer Support about the device chassis.
Assb ID or Assembly ID	Identification number that describes the FRU hardware.
FRU model number	Model number of FRU hardware component.
CLEI code	Common Language Equipment Identifier code. This value is displayed only for hardware components that use ID EEPROM format v2. This value is not displayed for components that use ID EEPROM format v1.
EEPROM Version	ID EEPROM version used by hardware component: 0x01 (version 1) or 0x02 (version 2).

Table 29: show chassis hardware Output Fields *(continued)*

Field Name	Field Description
Description	

Table 29: show chassis hardware Output Fields *(continued)*

Field Name	Field Description
	Brief description of the hardware item: • Type of power supply. • Switch Control Board (SCB) Starting with Junos OS Release 12.1X47-D15 and Junos OS Release 17.3R1, the SRX5K-SCBE (SCB2) is introduced. • There are three SCB slots in SRX5800 devices. The third slot can be used for an SCB or an FPC. When an SRX5K-SCB was used , the third SCB slot was used as an FPC. SCB redundancy is provided in chassis cluster mode. • With an SCB2, a third SCB is supported. If a third SCB is plugged in, it provides intra-chassis fabric redundancy. • The Ethernet switch in the SCB2 provides the Ethernet connectivity among all the FPCs and the Routing Engine. The Routing Engine uses this connectivity to distribute forwarding and routing tables to the FPCs. The FPCs use this connectivity to send exception packets to the Routing Engine. • Fabric connects all FPCs in the data plane. The Fabric Manager executes on the Routing Engine and controls the fabric system in the chassis. Packet Forwarding Engines on the FPC and fabric planes on the SCB are connected through HSL2 channels. • SCB2 supports HSL2 with both 3.11 Gbps and 6.22 Gbps (SerDes) link speed and various HSL2 modes. When an FPC is brought online, the link speed and HSL2 mode are determined by the type of FPC. Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the SRX5K-SCB3 (SCB3) with enhanced midplane is introduced. • All existing SCB software that is supported by SCB2 is supported on SCB3. • SRX5K-RE-1800X4 mixed Routing Engine use is not supported. • SCB3 works with the SRX5K-MPC (IOC2), SRX5K-MPC3-100G10G (IOC3), SRX5K-MPC3-40G10G (IOC3), and SRX5K-SPC-4-15-320 (SPC2) with current midplanes and the new enhanced midplanes. • Mixed SCB use is not supported. If an SCB2 and an SCB3 are used, the system will only power on the master Routing Engine's SCB and will power off the other SCBs. Only the SCB in slot 0 is powered on and a system log is generated. • SCB3 supports up to 400 Gbps per slot with old midplanes and up to 500 Gbps per slot with new midplanes. • SCB3 supports fabric intra-chassis redundancy. • SCB3 supports the same chassis cluster function as the SRX5K-SCB (SCB1) and the SRX5K-SCBE (SCB2), except for in-service software upgrade (ISSU)

Table 29: show chassis hardware Output Fields *(continued)*

Field Name	Field Description
	and in-service hardware upgrade (ISHU). • SCB3 has a second external Ethernet port. • Fabric bandwidth increasing mode is not supported. Starting in Junos OS 19.3R1, SRX5K-SCB4 is supported on SRX5600 and SRX5800 devices along with SRX5K-SPC3. SRX5K-SCB4: • Interoperate with SRX5K-RE3-128G, SRX5K-RE-1800X4, IOC2, IOC3, IOC4, SPC2, and SPC3. SCB4 is compatible with all midplanes and interoperate with existing PEMs, fan trays, and front panel displays. • Does not interoperate with SCB, SCB2, and SCB3. • Supports 480-Gbps link speed per slot. • Supports 1-Gigabit Ethernet interfaces speed with SRX5K-RE-1800X4 and 1-Gigabit, 2.5-Gigabit, and 10-Gigabit Ethernet speeds with SRX5K-RE3-128G. • Support ISHU and ISSU in chassis cluster. • Supports fabric bandwidth mode and redundant fabric mode on SRX5600 and SRX5800 devices. The bandwidth mode is the new default mode which is necessary to configure redundant mode in setting up the chassis cluster successfully.

Table 29: show chassis hardware Output Fields *(continued)*

Field Name	Field Description
	• Type of Flexible PIC Concentrator (FPC), Physical Interface Card (PIC), Modular Interface Cards (MICs), and PIMs.

Table 29: show chassis hardware Output Fields *(continued)*

Field Name	Field Description
	• IOCs Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the SRX5K-MPC3-100G10G (IOC3) and the SRX5K-MPC3-40G10G (IOC3) are introduced. • IOC3 has two types of IOC3 MPCs, which have different built-in MICs: the 24x10GE + 6x40GE MPC and the 2x100GE + 4x10GE MPC. • IOC3 supports SCB3 and SRX5000 line backplane and enhanced backplane. • IOC3 can only work with SRX5000 line SCB2 and SCB3. If an SRX5000 line SCB is detected, IOC3 is offline, an FPC misconfiguration alarm is raised, and a system log message is generated. • IOC3 interoperates with SCB2 and SCB3. • IOC3 interoperates with the SRX5K-SPC-4-15-320 (SPC2) and the SRX5K-MPC (IOC2). • The maximum power consumption for one IOC3 is 645W. An enhanced power module must be used. • The IOC3 does not support the following command to set a PIC to go offline or online: **request chassis pic fpc-slot <fpc-slot> pic-slot <pic-slot> <offline | online>** . • IOC3 supports 240 Gbps of throughput with the enhanced SRX5000 line backplane. • Chassis cluster functions the same as for the SRX5000 line IOC2. • IOC3 supports intra-chassis and inter-chassis fabric redundancy mode. • IOC3 supports ISSU and ISHU in chassis cluster mode. • IOC3 supports intra-FPC and and Inter-FPC Express Path (previously known as *services offloading*) with IPv4. • NAT of IPv4 and IPv6 in normal mode and IPv4 for Express Path mode. • All four PICs on the 24x10GE + 6x40GE cannot be powered on. A maximum of two PICs can be powered on at the same time. Use the **set chassis fpc <slot> pic <pic> power off** command to choose the PICs you want to power on. Fabric bandwidth increasing mode is not supported on IOC3. • SRX Clustering Module (SCM) • Fan tray Starting in Junos OS Release 19.3R1, the SRX5K-IOC4-10G and SRX5K-IOC4-MRAT line cards are supported along with SRX5K-SPC3 on the SRX5000 series devices.

Table 29: show chassis hardware Output Fields *(continued)*

Field Name	Field Description
	SRX5K-IOC4-10G: • Interoperates with SCB3, SCB4, SRX5K-RE-1800X4, SRX5K-RE3-128G, SPC2, SPC3, IOC2,IOC3, and IOC4. • Supports 480-Gbps speed. • Supports 40X10GE Interfaces with SCB3. • 40 10-Gigabit Ethernet port provides 10-Gigabit Ethernet MACsec support. • Supports reth and aggregated interfaces on the chassis cluster. • Supports ISSU and logical system on the chassis cluster. • Does not support SCB2. • SRX5K-IOC4-MRAT with SCB3 supports 10-Gigabit, 40-Gigabit, and 100-Gigabit Ethernet Interfaces. • For hosts, the Routing Engine type. Starting with Junos OS Release 12.1X47-D15 and Junos OS Release 17.3R1, the SRX5K-RE-1800X4 Routing Engine is introduced. • The SRX5K-RE-1800X4 has an Intel Quad core Xeon processor, 16 GB of DRAM, and a 128-GB solid-state drive (SSD). The number 1800 refers to the speed of the processor (1.8 GHz). The maximum required power for this Routing Engine is 90W. **NOTE:** The SRX5K-RE-1800X4 provides significantly better performance than the previously used Routing Engine, even with a single core. Starting in Junos OS Release 19.3R1, SRX5K-RE3-128G Routing Engine is supported along with SRX5K-SPC3 on the SRX5000 series devices. SRX5K-RE3-128G: • Provides improved control plane performance and scalability. SRX5K-RE3-128G has Intel's Haswell-EP based processor with six cores. • Supports two 200G SSDs to store log files and 128-GB of memory for storing routing and forwarding tables and for other routing engines. • Interoperates with SCB3, SCB4, SRX5K-RE3-128G, SPC2, SPC3, IOC2, IOC3, and IOC4. • Does not support SCB2 and SRX5K-RE-1800X4.

show chassis hardware

show chassis hardware (SRX5800)

user@host> **show chassis hardware**

```
node0:
--------------------------------------------------------------------------
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                JN1267B0FAGA      SRX5800
Midplane         REV 42   760-063937   ACRL3065          Enhanced SRX5800 Backpl

                            ane
FPM Board        REV 05   760-061272   CAHE4860          Front Panel Display
PDM              Rev 01   740-063049   QCS2209509D       Power Distribution Modu

                            le
PEM 0            Rev 04   740-034724   QCS171002016      PS 4.1kW; 200-240V AC i

                            n
PEM 1            Rev 11   740-027760   QCS1825N07S       PS 4.1kW; 200-240V AC i

                            n
Routing Engine 0 REV 01   750-095568   CALK8884          SRX5k RE-2000x6
Routing Engine 1 REV 01   750-095568   CADZ9076          SRX5k RE-2000x6
CB 0             REV 26   750-031391   CALV3002          SRX5k SCB4
CB 1             REV 26   750-031391   CALV3009          SRX5k SCB4
FPC 2            REV 28   750-073435   CALS4630          SPC3
  CPU                     BUILTIN      BUILTIN           SRX5k vCPP Broadwell
  PIC 0                   BUILTIN      BUILTIN           SPU Cp-Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
FPC 3            REV 17   750-044175   CABE7777          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
FPC 4            REV 08   750-061262   CAFD8147          SRX5k IOC II
  CPU            REV 02   711-061263   CAFV7488          SRX5k MPC PMB
  MIC 0          REV 03   750-055732   CAFV9369          20x 1GE(LAN) SFP
    PIC 0                 BUILTIN      BUILTIN           10x 1GE(LAN) SFP
      Xcvr 0     REV 02   740-011613   PNB1GJR           SFP-SX
    PIC 1                 BUILTIN      BUILTIN           10x 1GE(LAN) SFP
FPC 5            REV 10   750-062242   CAKX2328          SRX5k IOC3 2CGE+4XGE
```

```
  PIC 0                  BUILTIN      BUILTIN          2x 10GE SFP+
    Xcvr 0       REV 01   740-021308   ANA07RE          SFP+-10G-SR
    Xcvr 1       REV 01   740-031980   AQF0RBJ          SFP+-10G-SR
  PIC 1                  BUILTIN      BUILTIN          1x 100GE CFP2
  PIC 2                  BUILTIN      BUILTIN          2x 10GE SFP+
    Xcvr 0       REV 01   740-031980   AA1650304RF      SFP+-10G-SR
    Xcvr 1       REV 01   740-021308   AQ93BDK          SFP+-10G-SR
  PIC 3                  BUILTIN      BUILTIN          1x 100GE CFP2
FPC 8            REV 46   750-056519   CALC4514         SRX5k IOC4 MRAT
  CPU            REV 21   750-057177   CALC3494         SMPC PMB
  PIC 0                  BUILTIN      BUILTIN          MRATE-6xQSFPP-XGE-XLGE-CGE

    Xcvr 0       REV 01   740-059437   000T20128        QSFP28-LPBK
    Xcvr 1       REV 01   740-067443   1ACP13450KH      QSFP+-40G-SR4
  PIC 1                  BUILTIN      BUILTIN          MRATE-6xQSFPP-XGE-XLGE-CGE

    Xcvr 0       REV 01   740-059437   0000T3443        QSFP28-LPBK
Fan Tray 0       REV 06   740-035409   ACAE9390         Enhanced Fan Tray
Fan Tray 1       REV 06   740-035409   ACAE9386         Enhanced Fan Tray

node1:
--------------------------------------------------------------------------
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                JN1267B01AGA      SRX5800
Midplane         REV 42   760-063937   ACRL3068          Enhanced SRX5800 Backplane
FPM Board        REV 05   760-061272   CAJX9988          Front Panel Display
PDM              Rev 01   740-063049   QCS2209507A       Power Distribution Module
PEM 0            Rev 11   740-027760   QCS1822N0EY       PS 4.1kW; 200-240V AC in
PEM 1            Rev 03   740-034724   QCS17020203F      PS 4.1kW; 200-240V AC in
Routing Engine 0 REV 01   750-095568   CALK8904          SRX5k RE-2000x6
Routing Engine 1 REV 01   750-095568   CADZ9076          SRX5k RE-2000x6
CB 0             REV 26   750-031391   CALV3010          SRX5k SCB4
CB 1             REV 26   750-031391   CALV3000          SRX5k SCB4
FPC 2            REV 28   750-073435   CAKZ9620          SPC3
  CPU                     BUILTIN      BUILTIN           SRX5k vCPP Broadwell
  PIC 0                   BUILTIN      BUILTIN           SPU Cp-Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
FPC 3            REV 18   750-054877   CACH4082          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
```

```
FPC 4            REV 08   750-061262   CAFD8165          SRX5k IOC II
  CPU            REV 02   711-061263   CAFV7507          SRX5k MPC PMB
  MIC 0          REV 03   750-055732   CAFV6603          20x 1GE(LAN) SFP
    PIC 0                 BUILTIN      BUILTIN           10x 1GE(LAN) SFP
      Xcvr 0     REV 01   740-011613   AM0805S8M4N       SFP-SX
    PIC 1                 BUILTIN      BUILTIN           10x 1GE(LAN) SFP
FPC 5            REV 03   750-062242   CAFZ2748          SRX5k IOC3 2CGE+4XGE
  PIC 0                   BUILTIN      BUILTIN           2x 10GE SFP+
    Xcvr 0       REV 01   740-021308   11T511100788      SFP+-10G-SR
    Xcvr 1       REV 01   740-031980   AS92WJ0           SFP+-10G-SR
  PIC 1                   BUILTIN      BUILTIN           1x 100GE CFP2
  PIC 2                   BUILTIN      BUILTIN           2x 10GE SFP+
    Xcvr 0       REV 01   740-031980   AA1650304EZ       SFP+-10G-SR
    Xcvr 1       REV 01   740-031980   ANS0EAR           SFP+-10G-SR
  PIC 3                   BUILTIN      BUILTIN           1x 100GE CFP2
FPC 8            REV 46   750-056519   CALC4526          SRX5k IOC4 MRAT
  CPU            REV 21   750-057177   CALF5727          SMPC PMB
  PIC 0                  BUILTIN      BUILTIN          MRATE-6xQSFPP-XGE-XLGE-CGE

    Xcvr 1       REV 01   740-067443   1ACP13450L9       QSFP+-40G-SR4
  PIC 1                  BUILTIN      BUILTIN          MRATE-6xQSFPP-XGE-XLGE-CGE
Fan Tray 0       REV 06   740-035409   ACAE9298          Enhanced Fan Tray
Fan Tray 1       REV 06   740-035409   ACAE9314          Enhanced Fan Tray

{primary:node0}
```

show chassis hardware (SRX5600 and SRX5800 devices for SRX5K-MPC)

`user@host>` **show chassis hardware**

```
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                               JN12170EAAGA      SRX 5800
Midplane         REV 01   710-041799   ACAX3849          SRX 5800 Backplane
FPM Board        REV 01   710-024632   CAAX7297          Front Panel Display
PDM              Rev 03   740-013110   QCS170250DU       Power Distribution Modu
le
PEM 0            Rev 03   740-034724   QCS17020203F      PS 4.1kW; 200-240V AC i
n
PEM 1            Rev 03   740-034724   QCS17020203C      PS 4.1kW; 200-240V AC i
n
PEM 2            Rev 04   740-034724   QCS17100200A      PS 4.1kW; 200-240V AC i
n
PEM 3            Rev 03   740-034724   QCS17080200M      PS 4.1kW; 200-240V AC i
```

```
n
Routing Engine 0 REV 11   740-023530   9012047437        SRX5k RE-13-20
CB 0             REV 09   710-024802   CAAX7202          SRX5k SCB
CB 1             REV 09   710-024802   CAAX7157          SRX5k SCB
FPC 0            REV 07   750-044175   CAAD0791          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Cp
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
FPC 1            REV 07   750-044175   CAAD0751          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
FPC 2            REV 28   750-020751   CAAW1817          SRX5k DPC 4X 10GE
  CPU            REV 04   710-024633   CAAZ5269          SRX5k DPC PMB
  PIC 0                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
    Xcvr 0       REV 02   740-014289   T10A00404         XFP-10G-SR
  PIC 1                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
  PIC 2                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
  PIC 3                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
FPC 6            REV 02   750-044175   ZY2552            SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
FPC 9            REV 10   750-044175   CAAP5932          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
FPC 10           REV 22   750-043157   ZH8192            SRX5k IOC II   CPU
   REV 08   711-043360   YX3879           SRX5k MPC PMB
  MIC 0          REV 01   750-049488   YZ2084            10x 10GE SFP+
    PIC 0                 BUILTIN      BUILTIN           10x 10GE SFP+
      Xcvr 0     REV 01   740-031980   AMB0HG3           SFP+-10G-SR
      Xcvr 1     REV 01   740-031980   AM20B6F           SFP+-10G-SR
  MIC 1          REV 19   750-049486   CAAH3504          1x 100GE CFP
    PIC 2                 BUILTIN      BUILTIN           1x 100GE CFP
      Xcvr 0     REV 01   740-035329   X000D375          CFP-100G-SR10
FPC 11           REV 07.04.07 750-043157 CAAJ8771       SRX5k IOC II   CPU
   REV 08   711-043360   CAAJ3881           SRX5k MPC PMB
  MIC 0          REV 19   750-049486   CAAH0979          1x 100GE CFP
    PIC 0                 BUILTIN      BUILTIN           1x 100GE CFP
```

```
      Xcvr 0     REV 01   740-035329   UP1020Z          CFP-100G-SR10
  MIC 1          REV 08   750-049487   CAAM1160         2x 40GE QSFP+
    PIC 2                 BUILTIN      BUILTIN          2x 40GE QSFP+
      Xcvr 0     REV 01   740-032986   QB151094         QSFP+-40G-SR4
      Xcvr 1     REV 01   740-032986   QB160509         QSFP+-40G-SR4
Fan Tray 0       REV 04   740-035409   ACAE0875         Enhanced Fan Tray
Fan Tray 1       REV 04   740-035409   ACAE0876         Enhanced Fan Tray
```

show chassis hardware (with 20-Gigabit Ethernet MIC with SFP)

user@host> **show chassis hardware**

```
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                               JN108DA5AAGA      SRX 5800
Midplane         REV 02   710-013698   TR0037            SRX 5600 Midplane
FPM Board        REV 02   710-014974   JY4635            Front Panel Display
PDM              Rev 02   740-013110   QCS10465005       Power Distribution Module
PEM 0            Rev 03   740-023514   QCS11154040       PS 1.7kW; 200-240VAC in
PEM 2            Rev 02   740-023514   QCS10504014       PS 1.7kW; 200-240VAC in
Routing Engine 0 REV 05   740-015113   1000681023        RE-S-1300
CB 0             REV 05   710-013385   JY4775            SRX5k SCB
FPC 1            REV 17   750-020751   WZ6349            SRX5k DPC 4X 10GE
  CPU            REV 02   710-024633   WZ0718            SRX5k DPC PMB
  PIC 0                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
    Xcvr 0                NON-JNPR     C724XM088         XFP-10G-SR
  PIC 1                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
    Xcvr 0       REV 02   740-011571   C831XJ08S         XFP-10G-SR
  PIC 2                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
  PIC 3                   BUILTIN      BUILTIN           1x 10GE(LAN/WAN) RichQ
FPC 3            REV 22   750-043157   ZH8189            SRX5k IOC II
  CPU            REV 06   711-043360   YX3912            SRX5k MPC PMB
  MIC 0          REV 01   750-055732   CACF9115          20x 1GE(LAN) SFP
    PIC 0                 BUILTIN      BUILTIN           10x 1GE(LAN) SFP
      Xcvr 2     REV 02   740-013111   B358549           SFP-T
      Xcvr 9     REV 02   740-011613   PNB1FQS           SFP-SX
    PIC 1                 BUILTIN      BUILTIN           10x 1GE(LAN) SFP
      Xcvr 9     REV 02   740-011613   PNB1FFF           SFP-SX
FPC 5            REV 01   750-027945   JW9665            SRX5k FIOC
  CPU
FPC 8            REV 08   750-023996   XA7234            SRX5k SPC
  CPU            REV 02   710-024633   XA1599            SRX5k DPC PMB
  PIC 0                   BUILTIN      BUILTIN           SPU Cp-Flow
```

```
  PIC 1                 BUILTIN      BUILTIN          SPU Flow
Fan Tray 0       REV 03   740-014971   TP0902           Fan Tray
Fan Tray 1       REV 01   740-014971   TP0121           Fan Tray
```

show chassis hardware
(SRX5600 and SRX5800 devices with SRX5000 line SRX5K-SCBE [SCB2] and SRX5K-RE-1800X4 [RE2])

```
user@host> show chassis hardware
```

```
node0:
--------------------------------------------------------------------------
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                JN1251EA1AGB      SRX5600
Midplane         REV 01   760-063936   ACRE2657          Enhanced SRX5600 Midplane
FPM Board        REV 01   710-024631   CABY3551          Front Panel Display
PEM 0            Rev 03   740-034701   QCS13380901P      PS 1.4-2.6kW; 90-264V AC
 in
PEM 1            Rev 03   740-034701   QCS133809019      PS 1.4-2.6kW; 90-264V AC
 in
Routing Engine 0 REV 02   740-056658   9009210105        SRX5k RE-1800X4
Routing Engine 1 REV 02   740-056658   9013115551        SRX5k RE-1800X4
CB 0             REV 01   750-062257   CADW3663          SRX5k SCB3
CB 1             REV 01   750-062257   CADZ3263          SRX5k SCB3
FPC 0            REV 18   750-054877   CABG6043          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Cp
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
FPC 1            REV 01   750-062243   CAEE5918          SRX5k IOC3 24XGE+6XLG
  CPU            REV 02   711-062244   CADX8509          RMPC PMB
  PIC 0                   BUILTIN      BUILTIN           12x 10GE SFP+
    Xcvr 0       REV 01   740-031980   273363A01891      SFP+-10G-SR
    Xcvr 1       REV 01   740-031980   273363A01915      SFP+-10G-SR
    Xcvr 2       REV 01   740-031980   ANA0BK6           SFP+-10G-SR
    Xcvr 3       REV 01   740-031980   AP407GA           SFP+-10G-SR
    Xcvr 9       REV 01   740-021308   MUC20G1           SFP+-10G-SR
  PIC 1                   BUILTIN      BUILTIN           12x 10GE SFP+
  PIC 2                   BUILTIN      BUILTIN           3x 40GE QSFP+
  PIC 3                   BUILTIN      BUILTIN           3x 40GE QSFP+
  WAN MEZZ       REV 15   750-049136   CAEE5845          MPC5E 24XGE OTN Mezz
FPC 3            REV 11   750-043157   CACL7452          SRX5k IOC II
  CPU            REV 04   711-043360   CACP1977          SRX5k MPC PMB
```

```
  MIC 0          REV 04   750-049488   CABL4759          10x 10GE SFP+
    PIC 0                 BUILTIN      BUILTIN           10x 10GE SFP+
      Xcvr 0     REV 01   740-021308   CF36KM0SY         SFP+-10G-SR
      Xcvr 1     REV 01   740-021308   MUC0MF2           SFP+-10G-SR
      Xcvr 2     REV 01   740-021308   CF36KM01S         SFP+-10G-SR
      Xcvr 3     REV 01   740-021308   MUC229N           SFP+-10G-SR
FPC 5            REV 07   750-044175   CAAD0764          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
Fan Tray                                                 Enhanced Fan Tray

node1:
--------------------------------------------------------------------------
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                JN124FE77AGB      SRX5600
Midplane         REV 01   760-063936   ACRE2970          Enhanced SRX5600 Midplane
FPM Board        REV 01   710-024631   CABY3552          Front Panel Display
PEM 0            Rev 03   740-034701   QCS133809028      PS 1.4-2.6kW; 90-264V AC
 in
PEM 1            Rev 03   740-034701   QCS133809027      PS 1.4-2.6kW; 90-264V AC
 in
Routing Engine 0 REV 02   740-056658   9009218294        SRX5k RE-1800X4
Routing Engine 1 REV 02   740-056658   9013104758        SRX5k RE-1800X4
CB 0             REV 01   750-062257   CAEB8180          SRX5k SCB3
CB 1             REV 01   750-062257   CADZ3334          SRX5k SCB3
FPC 0            REV 18   750-054877   CACJ9834          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Cp
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
FPC 1            REV 01   750-062243   CAEB0981          SRX5k IOC3 24XGE+6XLG
  CPU            REV 02   711-062244   CAEA4644          RMPC PMB
  PIC 0                   BUILTIN      BUILTIN           12x 10GE SFP+
    Xcvr 0       REV 01   740-031980   AP41BLH           SFP+-10G-SR
    Xcvr 1       REV 01   740-031980   AQ400SL           SFP+-10G-SR
    Xcvr 2       REV 01   740-031980   AP422LJ           SFP+-10G-SR
    Xcvr 3       REV 01   740-021308   AMG0RBT           SFP+-10G-SR
    Xcvr 9       REV 01   740-021308   MUC2FRG           SFP+-10G-SR
  PIC 1                   BUILTIN      BUILTIN           12x 10GE SFP+
```

```
  PIC 2                   BUILTIN      BUILTIN          3x 40GE QSFP+
  PIC 3                   BUILTIN      BUILTIN          3x 40GE QSFP+
  WAN MEZZ       REV 15   750-049136   CAEA4837         MPC5E 24XGE OTN Mezz
FPC 3            REV 11   750-043157   CACA8784         SRX5k IOC II
  CPU            REV 04   711-043360   CACA8820         SRX5k MPC PMB
  MIC 0          REV 05   750-049488   CADF0521         10x 10GE SFP+
    PIC 0                 BUILTIN      BUILTIN          10x 10GE SFP+
      Xcvr 0     REV 01   740-030658   AD1130A00PV      SFP+-10G-USR
      Xcvr 1     REV 01   740-031980   AN40MVV          SFP+-10G-SR
      Xcvr 2     REV 01   740-021308   CF36KM37B        SFP+-10G-SR
      Xcvr 3     REV 01   740-021308   AD153830DSZ      SFP+-10G-SR
  MIC 1          REV 01   750-049487   CABB5961         2x 40GE QSFP+
    PIC 2                 BUILTIN      BUILTIN          2x 40GE QSFP+
      Xcvr 1     REV 01   740-032986   QB160513         QSFP+-40G-SR4
FPC 5            REV 02   750-044175   ZY2569           SRX5k SPC II
  CPU                     BUILTIN      BUILTIN          SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN          SPU Flow
  PIC 1                   BUILTIN      BUILTIN          SPU Flow
  PIC 2                   BUILTIN      BUILTIN          SPU Flow
  PIC 3                   BUILTIN      BUILTIN          SPU Flow
Fan Tray                                               Enhanced Fan Tray
```

show chassis hardware
(SRX5400, SRX5600, and SRX5800 devices with SRX5000 line SRX5K-SCB3 [SCB3] with enhanced midplanes and SRX5K-MPC3-100G10G [IOC3] or SRX5K-MPC3-40G10G [IOC3])

```
user@host> show chassis hardware
```

```
node0:
--------------------------------------------------------------------------
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                JN1250870AGB      SRX5600
Midplane         REV 01   760-063936   ACRE2578          Enhanced SRX5600 Midplane


FPM Board        REV 02   710-017254   KD9027            Front Panel Display
PEM 0            Rev 03   740-034701   QCS13090900T      PS 1.4-2.6kW; 90-264V A

                                       C in
PEM 1            Rev 03   740-034701   QCS13090904T      PS 1.4-2.6kW; 90-264V A

                                       C in
Routing Engine 0 REV 01   740-056658   9009196496        SRX5k RE-1800X4
```

```
CB 0             REV 01   750-062257   CAEC2501          SRX5k SCB3
FPC 0            REV 10   750-056758   CADC8067          SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Cp
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
FPC 2            REV 01   750-062243   CAEE5924          SRX5k IOC3 24XGE+6XLG
  CPU            REV 01   711-062244   CAEB4890          SRX5k IOC3 PMB
  PIC 0                   BUILTIN      BUILTIN           12x 10GE SFP+
  PIC 1                   BUILTIN      BUILTIN           12x 10GE SFP+
  PIC 2                   BUILTIN      BUILTIN           3x 40GE QSFP+
    Xcvr 0       REV 01   740-038623   MOC13156230449    QSFP+-40G-CU1M
    Xcvr 2       REV 01   740-038623   MOC13156230449    QSFP+-40G-CU1M
  PIC 3                   BUILTIN      BUILTIN           3x 40GE QSFP+
  WAN MEZZ       REV 01   750-062682   CAEE5817          24x 10GE SFP+ Mezz
FPC 4            REV 11   750-043157   CACY1595          SRX5k IOC II
  CPU            REV 04   711-043360   CACZ8879          SRX5k MPC PMB
  MIC 1          REV 04   750-049488   CACM6062          10x 10GE SFP+
    PIC 2                 BUILTIN      BUILTIN           10x 10GE SFP+
      Xcvr 7     REV 01   740-021308   AD1439301TU       SFP+-10G-SR
      Xcvr 8     REV 01   740-021308   AD1439301SD       SFP+-10G-SR
      Xcvr 9     REV 01   740-021308   AD1439301TS       SFP+-10G-SR
FPC 5            REV 05   750-044175   ZZ1371            SRX5k SPC II
  CPU                     BUILTIN      BUILTIN           SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN           SPU Flow
  PIC 1                   BUILTIN      BUILTIN           SPU Flow
  PIC 2                   BUILTIN      BUILTIN           SPU Flow
  PIC 3                   BUILTIN      BUILTIN           SPU Flow
Fan Tray                                                 Enhanced Fan Tray

node1:
--------------------------------------------------------------------------
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                JN124FEC0AGB      SRX5600
Midplane         REV 01   760-063936   ACRE2946          Enhanced SRX5600 Midplane
FPM Board        test     710-017254   test              Front Panel Display
PEM 0            Rev 01   740-038514   QCS114111003      DC 2.6kW Power Entry
Module
PEM 1            Rev 01   740-038514   QCS12031100J      DC 2.6kW Power Entry
Module
Routing Engine 0 REV 01   740-056658   9009186342        SRX5k RE-1800X4
CB 0             REV 01   750-062257   CAEB8178          SRX5k SCB3
```

```
FPC 0            REV 07   750-044175   CAAD0769         SRX5k SPC II
  CPU                     BUILTIN      BUILTIN          SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN          SPU Cp
  PIC 1                   BUILTIN      BUILTIN          SPU Flow
  PIC 2                   BUILTIN      BUILTIN          SPU Flow
  PIC 3                   BUILTIN      BUILTIN          SPU Flow
FPC 4            REV 11   750-043157   CACY1592         SRX5k IOC II
  CPU            REV 04   711-043360   CACZ8831         SRX5k MPC PMB
  MIC 1          REV 04   750-049488   CACN0239         10x 10GE SFP+
    PIC 2                 BUILTIN      BUILTIN          10x 10GE SFP+
      Xcvr 7     REV 01   740-031980   ARN23HW          SFP+-10G-SR
      Xcvr 8     REV 01   740-031980   ARN2FVW          SFP+-10G-SR
      Xcvr 9     REV 01   740-031980   ARN2YVM          SFP+-10G-SR
FPC 5            REV 10   750-056758   CADA8736         SRX5k SPC II
  CPU                     BUILTIN      BUILTIN          SRX5k DPC PPC
  PIC 0                   BUILTIN      BUILTIN          SPU Flow
  PIC 1                   BUILTIN      BUILTIN          SPU Flow
  PIC 2                   BUILTIN      BUILTIN          SPU Flow
  PIC 3                   BUILTIN      BUILTIN          SPU Flow
Fan Tray                                               Enhanced Fan Tray
```

show chassis hardware (SRX4200)

user@host> show chassis hardware

```
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                DK2816AR0020      SRX4200
Mainboard        REV 01   650-071675   16061032317       SRX4200
Routing Engine 0          BUILTIN      BUILTIN           SRX Routing Engine
FPC 0                     BUILTIN      BUILTIN           FEB
  PIC 0                   BUILTIN      BUILTIN           8x10G-SFP
    Xcvr 0       REV 01   740-038153   MOC11511530020    SFP+-10G-CU3M
    Xcvr 1       REV 01   740-038153   MOC11511530020    SFP+-10G-CU3M
    Xcvr 2       REV 01   740-038153   MOC11511530020    SFP+-10G-CU3M
    Xcvr 3       REV 01   740-038153   MOC11511530020    SFP+-10G-CU3M
    Xcvr 4       REV 01   740-021308   04DZ06A00364      SFP+-10G-SR
    Xcvr 5       REV 01   740-031980   233363A03066      SFP+-10G-SR
    Xcvr 6       REV 01   740-021308   AL70SWE           SFP+-10G-SR
    Xcvr 7       REV 01   740-031980   ALN0N6C           SFP+-10G-SR
```

```
    Xcvr 8       REV 01   740-030076   APF16220018NK1    SFP+-10G-CU1M
Power Supply 0   REV 04   740-041741   1GA26241849       JPSU-650W-AC-AFO
Power Supply 1   REV 04   740-041741   1GA26241846       JPSU-650W-AC-AFO
Fan Tray 0                                              SRX4200 0, Front to Back
 Airflow - AFO
Fan Tray 1                                              SRX4200 1, Front to Back
 Airflow - AFO
Fan Tray 2                                              SRX4200 2, Front to Back
 Airflow - AFO
Fan Tray 3                                              SRX4200 3, Front to Back
 Airflow - AFO
```

show chassis hardware (vSRX 3.0)

Starting in Junos OS Release 20.1R1, when vSRX 3.0 performs resource management, the vCPUs and RAM available to the instance are assigned based on what has been allocated prior to launching the instance. A maximum of 32 cores will be assigned to SRXPFE, for flow processing. Any allocation of cores in excess of 32 will automatically be assigned to the Routing Engine. For example, if 36 cores are allocated to the VM during the creation process, 32 cores are assigned for flow processing and 4 cores will be assigned to the RE. For memory allocations, up to 64G of vRAM would be used by the SRXPFE. Any allocated memory in excess of 64G would be assigned to system memory and would not be used for maintaining flow sessions information.

Table 30: Recommended vCPU and vRAM Combinations

vCPU Number	vRAM Size (G)
2	4
5	8
9	16
17	32

> **NOTE:** On a deployed vSRX, only memory scale up is supported. Scaling down memory on a deployed vSRX, is not supported. If you need to scale down memory, then a fresh install is required.

user@host> show chassis hardware

```
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                806dddb1a141      VSRX
Midplane
System IO
Routing Engine                                            VSRX-2CPU-8G memory
FPC 0                                                     FPC
  PIC 0                                                   VSRX DPDK GE
Power Supply 0
```

show chassis hardware clei-models

show chassis hardware clei-models
(SRX5600 and SRX5800 devices with SRX5000 line SRX5K-SCBE [SCB2] and SRX5K-RE-1800X4 [RE2])

`user@host>` **show chassis hardware clei-models node 1**

```
node1:
--------------------------------------------------------------------------
```

```
Hardware inventory:
Item             Version  Part number  CLEI code         FRU model number
Midplane         REV 01   710-024803                     SRX5800-BP-A
FPM Board        REV 01   710-024632                     SRX5800-CRAFT-A
PEM 0            Rev 04   740-034724                     SRX5800-PWR-4100-AC
PEM 1            Rev 05   740-034724                     SRX5800-PWR-4100-AC
Routing Engine 0 REV 01   740-056658   COUCATTBAA        SRX5K-RE-1800X4
CB 0             REV 01   750-056587   COUCATSBAA        SRX5K-SCBE
CB 1             REV 01   750-056587   COUCATSBAA        SRX5K-SCBE
CB 2             REV 01   750-056587   COUCATSBAA        SRX5K-SCBE
FPC 0            REV 18   750-054877   COUCATLBAA        SRX5K-SPC-4-15-320
  CPU                     BUILTIN
FPC 1            REV 18   750-054877   COUCATLBAA        SRX5K-SPC-4-15-320
  CPU                     BUILTIN
FPC 2            REV 18   750-054877   COUCATLBAA        SRX5K-SPC-4-15-320
  CPU                     BUILTIN
FPC 3            REV 11   750-043157   COUIBCWBAA        SRX5K-MPC
  MIC 0          REV 05   750-049486   COUIBCYBAA        SRX-MIC-1X100G-CFP
  MIC 1          REV 04   750-049488   COUIBCXBAA        SRX-MIC-10XG-SFPP
FPC 4            REV 18   750-054877   COUCATLBAA        SRX5K-SPC-4-15-320
  CPU                     BUILTIN
FPC 7            REV 18   750-054877   COUCATLBAA        SRX5K-SPC-4-15-320
  CPU                     BUILTIN
FPC 8            REV 11   750-043157   COUIBCWBAA        SRX5K-MPC
  MIC 0          REV 05   750-049486   COUIBCYBAA        SRX-MIC-1X100G-CFP
FPC 9            REV 18   750-054877   COUCATLBAA        SRX5K-SPC-4-15-320
  CPU                     BUILTIN
FPC 10           REV 18   750-054877   COUCATLBAA        SRX5K-SPC-4-15-320
  CPU                     BUILTIN
Fan Tray 0       REV 04   740-035409                     SRX5800-HC-FAN
Fan Tray 1       REV 04   740-035409                     SRX5800-HC-FAN
```

show chassis pic (Security)

Syntax

```
show chassis pic fpc-slot slot-number pic-slot slot-number
```

Release Information

Command introduced in Junos OS Release 9.2.

Description

Display status information about the PIC installed in the specified Flexible PIC Concentrator (FPC) and PIC slot.

Options

fpc-slot *slot-number*—Display information about the FPC in the slot.

pic-slot *slot-number*—Display information about the PIC in this particular FPC slot.

Required Privilege Level

view

RELATED DOCUMENTATION

> *Interfaces User Guide for Security Devices*
> Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show chassis pic fpc-slot pic-slot on page 500

Output Fields

Table 31 on page 499 lists the output fields for the **show chassis pic** command. Output fields are listed in the approximate order in which they appear.

Table 31: show chassis pic Output Fields

Field Name	Field Description
Type	PIC type.

Table 31: show chassis pic Output Fields *(continued)*

Field Name	Field Description
State	Status of the PIC. State is displayed only when a PIC is in the slot. • **Online**— PIC is online and running. • **Offline**—PIC is powered down.
PIC version	PIC hardware version.
Uptime	How long the PIC has been online.
Port Number	Port number for the PIC.
Cable Type	Type of cable connected to the port: **LH**, **LX**, or **SX.**
PIC Port Information	Port-level information for the PIC. • Port—Port number • Cable type—Type of transceiver installed. • Fiber type—Type of fiber. • Xcvr vendor—Transceiver vendor name. • Xcvr vendor part number—Transceiver vendor part number. • Wavelength—Wavelength of the transmitted signal.

Sample Output

show chassis pic fpc-slot pic-slot

```
user@host> show chassis pic fpc-slot 10 pic-slot 0
```

```
FPC slot 10, PIC slot 0 information:
  Type                            10x 10GE SFP+
  State                           Online
  PIC version                  1.1
  Uptime                          6 days, 7 hours, 29 minutes, 28 seconds

PIC port information:
                         Fiber                    Xcvr vendor
  Port  Cable type       type  Xcvr vendor        part number       Wavelength
  0     10GBASE SR       MM    FINISAR CORP.      FTLX8571D3BNL-J1  850 nm
```

```
        Xcvr vendor
        firmware version
        0.0

PIC port information:
                          Fiber                    Xcvr vendor
  Port  Cable type        type  Xcvr vendor        part number       Wavelength
  1     10GBASE SR        MM    FINISAR CORP.      FTLX8571D3BNL-J1  850 nm

        Xcvr vendor
        firmware version
        0.0
```

show chassis power

Syntax

```
show chassis power
```

Release Information

Command modified in Junos OS Release 12.1X44-D10.

Description

Display power limits and usage information for the Power Entry Modules (PEMs).

Options

node— Displays information specific to the chassis.

sequence— Shows the chassis fru power on sequence.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

show chassis power sequence | 507

List of Sample Output

show chassis power on page 503

show chassis power detail on page 505

Output Fields

Table 32 on page 503 lists the output fields for the **show chassis power** command. Output fields are listed in the approximate order in which they appear.

Table 32: show chassis power Output Fields

Field Name	Field Description

PEM number	AC or DC PEM number on the chassis. The following output fields are displayed for the PEM:
	• State—State of the PEM:
	  ◦ Online—PEM is present in the slot and online.
	  ◦ Empty—PEM is not present in the slot.
	  ◦ Present—PEM is present in the slot, but not online.
	• AC Input —State of the AC input power feed with the number of active and expected feeds (1 or 2).
	• Capacity—Actual power input capacity with maximum capacity displayed (in parentheses) in watts.
	• DC Output—DC power output, in watts, for the specified zone, at the specified amps and voltage (A @ V), and load and percentage utilization of the maximum capacity for the zone.

System	Overall power statistics for the system zone:
	• Zone number:
	  ◦ Capacity—Maximum power capacity available for the zone, in watts.
	  ◦ Allocated power—Actual capacity allocated for the zone, in watts, with remaining power displayed in parentheses.
	  ◦ Actual usage—Actual power usage for the zone, in watts.
	• Total system capacity—Cumulative power capacity of all the zones, in watts.
	• Total remaining capacity—Difference between the total system capacity and cumulative allocated power of all zones, in watts.

Sample Output

show chassis power

When you enter the **show chassis power** command, the sample output is shown for DC PEM.

```
user@host> show chassis power
```

```
node0:
--------------------------------------------------------------------------

PEM 0:
```

```
  State:     Online
  DC input:  OK (1 feed expected, 1 feed connected)
  Capacity:  2440 W (maximum 2440 W)
  DC output: 600 W (zone 0, 12 A at 50 V, 24% of capacity)

PEM 1:
  State:     Online
  DC input:  OK (1 feed expected, 1 feed connected)
  Capacity:  2440 W (maximum 2440 W)
  DC output: 1550 W (zone 1, 31 A at 50 V, 63% of capacity)

PEM 2:
  State:     Empty
  Input:     Absent

PEM 3:
  State:     Empty
  Input:     Absent

System:
  Zone 0:
      Capacity:          2440 W (maximum 2440 W)
      Allocated power:   1050 W (1390 W remaining)
      Actual usage:      600 W
  Zone 1:
      Capacity:          2440 W (maximum 2440 W)
      Allocated power:   2310 W (130 W remaining)
      Actual usage:      1550 W
  Total system capacity: 4880 W (maximum 4880 W)
  Total remaining power: 1520 W

node1:
--------------------------------------------------------------------------------

PEM 0:
  State:     Online
  DC input:  OK (1 feed expected, 1 feed connected)
  Capacity:  2440 W (maximum 2440 W)
  DC output: 600 W (zone 0, 12 A at 50 V, 24% of capacity)

PEM 1:
  State:     Online
  DC input:  OK (1 feed expected, 1 feed connected)
  Capacity:  2440 W (maximum 2440 W)
```

```
  DC output: 1519 W (zone 1, 31 A at 49 V, 62% of capacity)

PEM 2:
  State:     Empty
  Input:     Absent

PEM 3:
  State:     Empty
  Input:     Absent

System:
  Zone 0:
      Capacity:          2440 W (maximum 2440 W)
      Allocated power:   1050 W (1390 W remaining)
      Actual usage:      600 W
  Zone 1:
      Capacity:          2440 W (maximum 2440 W)
      Allocated power:   2310 W (130 W remaining)
      Actual usage:      1519 W
  Total system capacity: 4880 W (maximum 4880 W)
  Total remaining power: 1520 W
```

show chassis power detail

When you enter the **show chassis power detail** command, the sample output is shown for DC PEM.

`user@host>` **show chassis power detail**

```
PEM 0:
  State:     Online
  Capacity:  1600 W (maximum 1600 W)
  DC output: 240 W (zone 0, 20 A at 12 V, 8% of capacity)

PEM 1:
  State:     Online
  Capacity:  1600 W (maximum 1600 W)
  DC output: 204 W (zone 0, 17 A at 12 V, 7% of capacity)

System:
  Zone 0:
      Capacity:          3200 W (maximum 3200 W)
      Allocated power:   1000 W (2200 W remaining)
      Actual usage:      444 W
  Total system capacity: 3200 W (maximum 3200 W)
```

```
  Total remaining power: 2200 W

Item                 Used(W)
  Fan Tray 0              0
  Fan Tray 1              0
  Fan Tray 2              0
  Fan Tray 3              0
  Fan Tray 4              0
  RE0/CB0                 0
  FPC 0                   0
  FPC 1                   0
```

show chassis power sequence

Syntax

```
show chassis power sequence
```

Release Information

Command modified in Junos OS Release 12.1X44-D10.

Description

Display the power-on sequence for the FPCs in the chassis. The numbers indicate the slot number of the FPCs. This document is supported on the SRX5400, SRX5600, and SRX5800 devices.

Options

This command has no options.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

fru-poweron-sequence | 327

List of Sample Output

show chassis power sequence on page 508

Output Fields

Table 33 on page 507 lists the output fields for the **show chassis power sequence** command. Output fields are listed in the approximate order in which they appear.

Table 33: show chassis power sequence Output Fields

Field Name	Field Description
Chassis FRU Power Sequence	Power-on sequence for the FPCs in the chassis. The numbers indicate the slot number of the FPCs.

Sample Output

show chassis power sequence

```
user@host> show chassis power sequence
```

```
Chassis FRU Power On Sequence: 0 1 2 3 4 5 6 7 8 9 10 11
```

show firewall (View)

Syntax

```
show firewall
<filter filter-name>
<counter counter-name>
<log>
<prefix-action-stats>
<terse>
```

Release Information
Command introduced before Junos OS Release 10.0 .

Description
Display statistics about configured firewall filters.

Options
none—Display statistics about configured firewall filters.

filter *filter-name*—Name of a configured filter.

counter *counter-name*—Name of a filter counter.

log—Display log entries for firewall filters.

prefix-action-stats—Display prefix action statistics for firewall filters.

terse—Display firewall filter names only.

Required Privilege Level
view

RELATED DOCUMENTATION

firewall

List of Sample Output
show firewall on page 510

Output Fields
Table 34 on page 510 lists the output fields for the **show firewall** command. Output fields are listed in the approximate order in which they appear.

Table 34: show firewall Output Fields

Field Name	Field Description
Filter	Name of a filter that has been configured with the **filter** at the **[edit firewall]** hierarchy level. When an interface-specific filter is displayed, the name of the filter is followed by the full interface name and by either **-i** for an input filter or **-o** for an output filter. When dynamic filters are displayed, the name of the filter is followed by the full interface name and by either **-in** for an input filter or **-out** for an output filter. When a logical system–specific filter is displayed, the name of the filter is prefixed with two underscore (__) characters and the name of the logical system (for example, **__ls1/filter1**).
Counters	Display filter counter information: • **Name**—Name of a filter counter that has been configured with the **counter** firewall filter action. • **Bytes**—Number of bytes that match the filter term under which the **counter** action is specified. • **Packets**—Number of packets that matched the filter term under which the **counter** action is specified.
Policers	Display policer information: • **Name**—Name of policer. • **Bytes**—Number of bytes that match the filter term under which the policer action is specified. This is only the number out-of-specification (out-of-spec) byte counts, not all the bytes in all packets policed by the policer. • **Packets**—Number of packets that matched the filter term under which the policer action is specified. This is only the number of out-of-specification (out-of-spec) packet counts, not all packets policed by the policer.

Sample Output

show firewall

```
user@host> show firewall
```

```
Filter: ef_path
Counters:
Name                                            Bytes              Packets
def-count                                       0                    0
video-count                                     0                    0
```

```
voice-count                              0                    0

Filter: __default_bpdu_filter__

Filter: deep
Counters:
Name                                     Bytes              Packets
deep2                                    302076               5031

Filter: deep-flood
Counters:
Name                                     Bytes              Packets
deep_flood_def                           302136               5032
deep1                                    0                    0
Policers:
Name                                               Packets
deep-pol-op-first                                  0
```

show interfaces (View Aggregated Ethernet)

Syntax

```
show interfaces <aenumber | rethnumber >
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index snmp-index>
<statistics>
```

Release Information

Command modified in Junos OS Release 10.2.

Description

Display status information about the specified aggregated Ethernet interface or redundant Ethernet interface. If you do not specify an interface name, status information for all interfaces is displayed.

> **NOTE:** This command only provides interface statistics for a redundant ethernet interface (reth) when executed on the node which has the active members/links of the redundant ethernet interface.

Options

- ae*number* | reth*number*—(Optional) Display standard information about the specified aggregated Ethernet interface or redundant Ethernet interface.
- brief | detail | extensive | terse—(Optional) Display the specified level of output.
- descriptions—(Optional) Display interface description strings.
- media—(Optional) Display media-specific information.
- snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.
- statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show interfaces extensive (Aggregated Ethernet) on page 521

Output Fields

Table 35 on page 513 lists the output fields for the **show interfaces** (Aggregated Ethernet) command. Output fields are listed in the approximate order in which they appear.

Table 35: Aggregated Ethernet show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface and state of the interface.	All levels
Enabled	State of the physical interface.	All levels
Interface index	Index number of the physical interface, which reflects its initialization sequence.	All levels
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Link-level type	Encapsulation being used on the physical interface.	All levels
MTU	Maximum transmission unit size on the physical interface.	All levels
Speed	Speed at which the interface is running.	All levels
Loopback	Loopback status: **Enabled** or **Disabled**. If loopback is enabled, type of loopback: **Local** or **Remote**.	All levels
Source filtering	Source filtering status: **Enabled** or **Disabled**.	All levels
Flow control	Flow control status: **Enabled** or **Disabled**.	All levels
Minimum links needed	Number of child links that must be operational for the aggregate interface to be operational.	All levels
Device flags	Information about the physical device.	All levels
Interface flags	Information about the interface.	All levels
Current address	Configured MAC address.	**detail extensive**

Table 35: Aggregated Ethernet show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Hardware address	Hardware MAC address.	**detail extensive**
Last flapped	Date, time, and how long ago the interface went from down to up or up to down. The format is **Last flapped: *year-month-day hours:minutes:seconds timezone* (*hours:minutes:seconds* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).**	**detail extensive**
Input Rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output Rate	Output rate in bps and pps.	None specified
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **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.	**detail extensive**
IPv6 transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the physical interface if IPv6 statistics tracking is enabled. • **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.	**detail extensive**

Table 35: Aggregated Ethernet show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Input errors	Input errors on the interface: • **Errors**—Sum of incoming frame aborts and frame check sequence (FCS) errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Framing errors**—Number of packets received with an invalid FCS. • **Runts**—Number of frames received that are smaller than the runt threshold. • **Giants**—Number of frames received that are larger than the giant threshold. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that the Junos OS does not handle. • **Resource errors**—Sum of transmit drops.	**detail extensive**
Output errors	Output errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious: • **Carrier transitions** —Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **MTU errors**—Number of packets whose size exceeded the MTU of the interface. • **Resource errors**—Sum of transmit drops.	**detail extensive**
Egress queues	Total number of egress queues supported on the specified interface	**detail extensive**

Table 35: Aggregated Ethernet show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Queue counters	CoS queue number and its associated user-configured forwarding class name. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism.	**detail extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Index number of the logical interface (which reflects its initialization sequence).	**detail extensive** none
SNMP ifIndex	SNMP interface index number of the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface.	All levels
VLAN-Tag	Tag Protocol Identifier (TPID) and VLAN identifier.	All levels
Demux	IP demultiplexing (demux) value that appears if this interface is used as the demux underlying interface. The output is one of the following: • Source Family Inet • Destination Family Inet	**detail extensive** none
Encapsulation	Encapsulation on the logical interface.	All levels
Statistics	Information about the number of packets, packets per second, number of bytes, and bytes per second on this aggregate interface. • **Bundle**—Information about input and output bundle rates. • **Link**—(**detail** and **extensive** only) Information about specific links in the aggregate, including link state and input and output rates.	**detail extensive** none

Table 35: Aggregated Ethernet show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output

LACP info	Link Aggregation Control Protocol (LACP) information for each aggregated interface.
	• **Role** can be one of the following:
	  ◦ **Actor**—Local device participating in LACP negotiation.
	  ◦ **Partner**—Remote device participating in LACP negotiation.
	• **System priority**—Priority assigned to the system (by management or administrative policy), encoded as an unsigned integer.
	• **System identifier**—Actor or partner system ID, encoded as a MAC address.
	• **Port priority**—Priority assigned to the port by the actor or partner (by management or administrative policy), encoded as an unsigned integer.
	• **Port number**—Port number assigned to the port by the actor or partner, encoded as an unsigned integer.
	• **Port key**—Operational key value assigned to the port by the actor or partner, encoded as an unsigned integer.
		detail extensive
		none

LACP Statistics	LACP statistics for each aggregated interface. • **LACP Rx**—LACP received counter that increments for each normal hello. • **LACP Tx**—Number of LACP transmit packet errors logged. • **Unknown Rx**—Number of unrecognized packet errors logged. • **Illegal Rx**—Number of invalid packets received.	**detail extensive** none
Marker Statistic	(**detail** and **extensive** only) Information about 802.3ad marker protocol statistics on the specified links. • **Marker Rx**—Number of valid marker PDUs received on this aggregation port. • **Resp Tx**—Number of marker response PDUs transmitted on this aggregation port. • **Unknown Rx**—Number of frames received that either carry the slow protocols Ethernet type value (43B.4) but contain an unknown protocol data unit (PDU), or are addressed to the slow protocols group MAC address (43B.3) but do not carry the slow protocols Ethernet type. • **Illegal Rx**—Number of frames received that carry the slow protocols Ethernet type value (43B.4) but contain a badly formed PDU or an illegal value of protocol subtype (43B.4).	**detail extensive** none

Table 35: Aggregated Ethernet show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Flow Statistics	Flow statistics for each aggregated interface.	**detail extensive** none
Flow Input statistics	Statistics for packets received by the flow module.	**detail extensive** none
Flow Output statistics	Statistics for packets sent by the flow module.	**detail extensive** none

Flow error statistics (Packets dropped due to)	Packet drop statistics for the flow module.
	• **Address spoofing**—Packet dropped when the screen module detected address spoofing.
	• **Authentication failed**—Packet dropped because the IPsec Encapsulating Security Payload (ESP) or Authentication Header (AH) authentication failed.
	• **Incoming NAT errors**—Packet dropped because the source NAT rule search failed, an invalid source NAT binding was found, or the NAT allocation failed.
	• **Invalid zone received packet**—This counter is not currently in use.
	• **Multiple user authentications**—Packet dropped if it matches more than one policy that specifies user authentication. (Sometimes packets are looped through the system more than once. Each time a packet passes through the system, that packet must be permitted by a policy.)
	• **Multiple incoming NAT**—Packet dropped if source NAT is specified more than once. (Sometimes packets are looped through the system more than once.)
	• **No parent for a gate**—This counter is not currently in use.
	• **No one interested in self packets**—This counter is incremented for one of the following reasons:
	  ◦ The outbound interface is a self interface, but the packet is not marked as a to-self packet and the destination address is in a source NAT pool.
	  ◦ No service is interested in the to-self packet
	  ◦ When a zone has ident-reset service enabled, the TCP RST to IDENT request for port 113 is sent back and this counter is incremented.
		detail extensive
		none

Table 35: Aggregated Ethernet show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output

Flow error statistics (Packets dropped due to)	Packet drop statistics for the flow module (continued).
	• **No minor session**—Packet dropped because no minor sessions are available and a minor session was requested. Minor sessions are allocated for storing additional TCP state information.
	• **No more sessions**—Packet dropped because there were no more free sessions available.
	• **No NAT gate**—This counter is not currently in use.
	• **No route present**—Packet dropped because a valid route was not available to forward the packet.
	• **No SA for incoming SPI**—Packet dropped because the incoming IPsec packet's security parameter index (SPI) does not match any known SPI.
	• **No tunnel found**—Packet dropped because a valid tunnel could not be found.
	• **No session for a gate**—Packet dropped by an ALG.
	• **No zone or NULL zone binding**—Packet dropped because its incoming interface was not bound to any zone.
	• **Policy denied**—The error counter is incremented for one of the following reasons:
	  ◦ Source or destination NAT (or both) has occurred and policy says to drop the packet.
	  ◦ Policy specifies user authentication, which failed.
	  ◦ Policy was configured to deny this packet.
	• **Security association not active**—Packet dropped because an IPsec packet was received for an inactive SA.
	• **TCP sequence number out of window**—TCP packet with a sequence number failed the TCP sequence number check that was received.
	• **Syn-attack protection**—Packet dropped because of SYN attack protection or SYN cookie protection.
	• **User authentication errors**—Packet dropped because policy requires authentication; however:
	  ◦ Only Telnet, FTP, and HTTP traffic can be authenticated.
	  ◦ The corresponding authentication entry could not be found, if web-auth is specified.
	  ◦ The maximum number of authenticated sessions per user was exceeded.
		detail extensive
		none

protocol-family	Protocol family configured on the logical interface.	**brief**

Table 35: Aggregated Ethernet show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Protocol	Protocol family configured on the logical interface.	**detail extensive** none
MTU	Maximum transmission unit size on the logical interface.	**detail extensive** none
Maximum labels	Maximum number of MPLS labels configured for the MPLS protocol family on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route Table	Routing table in which the logical interface address is located. For example, **0** refers to the routing table inet.0.	**detail extensive**
Flags	Information about protocol family flags.	**detail extensive** none
Mac-Validate Failures	Number of MAC address validation failures for packets and bytes. This field is displayed when MAC address validation is enabled for the logical interface.	**detail extensive** none
Addresses, Flags	Information about address flags.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address of the logical interface.	**detail extensive** none
Policer	Policer to be evaluated when packets are received or transmitted on the interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Sample Output

show interfaces extensive (Aggregated Ethernet)

`user@host>` **show interfaces ae0 extensive**

```
Physical interface: ae0, Enabled, Physical link is Up
  Interface index: 1973, SNMP ifIndex: 501, Generation: 2176
  Link-level type: Ethernet, MTU: 1518, Speed: 3Gbps, BPDU Error: None, MAC-REWRITE
 Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Disabled, Minimum links needed: 1,
Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Current address: 00:1f:12:8c:af:c0, Hardware address: 00:1f:12:8c:af:c0
  Last flapped   : 2010-04-16 14:25:36 PDT (00:02:50 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   64                   0 bps
   Output bytes  :           9816525824            463779840 bps
   Input  packets:                    1                   0 pps
   Output packets:             38345804              226455 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Dropped traffic statistics due to STP State:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
 0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort             38270790             38270790                    0
    1 expedited-fo                   0                    0                    0
    2 assured-forw                   0                    0                    0
    3 network-cont                 526                  526                    0
```

```
  Logical interface ae0.0 (Index 69) (SNMP ifIndex 502) (Generation 692)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.11 ]  Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :             1          0            64            0
        Output:      38572259     226453    9874497884    463775744
    Link:
      ge-5/0/1.0
        Input :             0          0             0            0
        Output:      12743866      75484    3262429696    154591232
      ge-5/2/0.0
        Input :             1          0            64            0
        Output:      13043256      75484    3339073116    154591232
      ge-5/2/1.0
        Input :             0          0             0            0
        Output:      12785137      75485    3272995072    154593280
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-5/0/1.0                0           0            0            0
      ge-5/2/0.0                0           0            0            0
      ge-5/2/1.0                0           0            0            0
    Security: Zone: HOST
   Allowed host-inbound traffic : bootp bfd bgp dns dvmrp igmp ldp msdp nhrp ospf
 pgm pim rip router-discovery rsvp sap vrrp dhcp
    finger ftp tftp ident-reset http https ike netconf ping reverse-telnet
reverse-ssh rlogin rpm rsh snmp snmp-trap ssh telnet
    traceroute xnm-clear-text xnm-ssl lsping ntp sip
    Flow Statistics :
    Flow Input statistics :
      Self packets :                    0
      ICMP packets :                    0
      VPN packets :                     0
      Multicast packets :               0
      Bytes permitted by policy :       0
      Connections established :         0
    Flow Output statistics:
      Multicast packets :               0
      Bytes permitted by policy :       8976842784
    Flow error statistics (Packets dropped due to):
      Address spoofing:                 0
      Authentication failed:            0
      Incoming NAT errors:              0
      Invalid zone received packet:     0
      Multiple user authentications:    0
      Multiple incoming NAT:            0
```

```
      No parent for a gate:              0
      No one interested in self packets: 0
      No minor session:                  0
      No more sessions:                  0
      No NAT gate:                       0
      No route present:                  0
      No SA for incoming SPI:            0
      No tunnel found:                   0
      No session for a gate:             0
      No zone or NULL zone binding       0
      Policy denied:                     0
      Security association not active:   0
      TCP sequence number out of window: 0
      Syn-attack protection:             0
      User authentication errors:        0
    Protocol inet, MTU: 1500, Generation: 841, Route table: 0
      Flags: Sendbcast-pkt-to-re
      Addresses, Flags: Is-Preferred Is-Primary
       Destination: 10.1.1/24, Local: 10.1.1.2, Broadcast: 10.1.1.255, Generation:
422
    Protocol multiservice, MTU: Unlimited, Generation: 842, Route table: 0
      Flags: Is-Primary
      Policer: Input: __default_arp_policer__
  Logical interface ae0.32767 (Index 83) (SNMP ifIndex 503) (Generation 693)
    Flags: SNMP-Traps 0x4004000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :             0          0             0            0
        Output:             0          0             0            0
    Link:
      ge-5/0/1.32767
        Input :             0          0             0            0
        Output:             0          0             0            0
      ge-5/2/0.32767
        Input :             0          0             0            0
        Output:             0          0             0            0
      ge-5/2/1.32767
        Input :             0          0             0            0
        Output:             0          0             0            0
    LACP info:        Role     System             System      Port    Port  Port

                              priority          identifier  priority  number   key

      ge-5/0/1.32767    Actor      127 00:1f:12:8c:af:c0       127     833     1
```

```
      ge-5/0/1.32767  Partner      127 00:1f:12:8f:d7:c0       127     641     1
      ge-5/2/0.32767    Actor      127 00:1f:12:8c:af:c0       127     848     1
      ge-5/2/0.32767  Partner      127 00:1f:12:8f:d7:c0       127     656     1
      ge-5/2/1.32767    Actor      127 00:1f:12:8c:af:c0       127     849     1
      ge-5/2/1.32767  Partner      127 00:1f:12:8f:d7:c0       127     657     1
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-5/0/1.32767           342         511            0            0
      ge-5/2/0.32767           344         498            0            0
      ge-5/2/1.32767           344         500            0            0
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-5/0/1.32767             0           0            0            0
      ge-5/2/0.32767             0           0            0            0
      ge-5/2/1.32767             0           0            0            0
    Security: Zone: HOST
    Allowed host-inbound traffic : bootp bfd bgp dns dvmrp igmp ldp msdp nhrp ospf
 pgm pim rip router-discovery rsvp sap vrrp dhcp
    finger ftp tftp ident-reset http https ike netconf ping reverse-telnet
reverse-ssh rlogin rpm rsh snmp snmp-trap ssh telnet
    traceroute xnm-clear-text xnm-ssl lsping ntp sip
    Flow Statistics :
    Flow Input statistics :
      Self packets :                    0
      ICMP packets :                    0
      VPN packets :                     0
      Multicast packets :               0
      Bytes permitted by policy :       0
      Connections established :         0
    Flow Output statistics:
      Multicast packets :               0
      Bytes permitted by policy :       0
    Flow error statistics (Packets dropped due to):
      Address spoofing:                 0
      Authentication failed:            0
      Incoming NAT errors:              0
      Invalid zone received packet:     0
      Multiple user authentications:    0
      Multiple incoming NAT:            0
      No parent for a gate:             0
      No one interested in self packets: 0
      No minor session:                 0
      No more sessions:                 0
      No NAT gate:                      0
      No route present:                 0
      No SA for incoming SPI:           0
```

```
    No tunnel found:                   0
    No session for a gate:             0
    No zone or NULL zone binding       0
    Policy denied:                     0
    Security association not active:   0
    TCP sequence number out of window: 0
    Syn-attack protection:             0
    User authentication errors:        0
  Protocol multiservice, MTU: Unlimited, Generation: 843, Route table: 0
    Flags: None
    Policer: Input: __default_arp_policer__
```

show interfaces diagnostics optics

Syntax

```
show interfaces diagnostics optics interface-name
```

Release Information

Command introduced in Junos OS Release 10.1.

Description

Display diagnostics data and alarms for Gigabit Ethernet optical transceivers (SFP) installed in SRX Series Services Gateways. The information provided by this command is known as digital optical monitoring (DOM) information.

Thresholds that trigger a high alarm, low alarm, high warning, or low warning are set by the transponder vendors. Generally, a high alarm or low alarm indicates that the optics module is not operating properly. This information can be used to diagnose why a transceiver is not working.

> **NOTE:** In a chassis cluster, the **show interfaces diagnostics optics** command works only on the node that is primary in redundancy group 0 (RG0).

Options

interface-name—Name of the interface associated with the port in which the transceiver is installed:

ge-*fpc/pic/port* .

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Interfaces

List of Sample Output

show interfaces diagnostics optics on page 530

Output Fields

Table 36 on page 527 lists the output fields for the show interfaces diagnostics optics command. Output fields are listed in the general order in which they appear.

Table 36: show interfaces diagnostics optics Output Fields

Field Name	Field Description
Physical interface	Displays the name of the physical interface.
Laser bias current	Displays the magnitude of the laser bias power setting current, in milliamperes. The laser bias provides direct modulation of laser diodes and modulates currents.
Laser output power	Displays the laser output power, in milliwatts (mW) and decibels referred to 1.0 mW (dBm).
Module temperature	Displays the temperature, in Celsius and Fahrenheit.
Module voltage	Displays the voltage, in Volts.
Receiver signal average optical power	Displays the receiver signal average optical power, in milliwatts (mW) and decibels referred to 1.0 mW (dBm).
Laser bias current high alarm	Displays whether the laser bias power setting high alarm is **On** or **Off**.
Laser bias current low alarm	Displays whether the laser bias power setting low alarm is **On** or **Off**.
Laser bias current high warning	Displays whether the laser bias power setting high warning is **On** or **Off**.
Laser bias current low warning	Displays whether the laser bias power setting low warning is **On** or **Off**.
Laser output power high alarm	Displays whether the laser output power high alarm is **On** or **Off**.
Laser output power low alarm	Displays whether the laser output power low alarm is **On** or **Off**.
Laser output power high warning	Displays whether the laser output power high warning is **On** or **Off**.
Laser output power low warning	Displays whether the laser output power low warning is **On** or **Off**.
Module temperature high alarm	Displays whether the module temperature high alarm is **On** or **Off**.
Module temperature low alarm	Displays whether the module temperature low alarm is **On** or **Off**.

Table 36: show interfaces diagnostics optics Output Fields *(continued)*

Field Name	Field Description
Module temperature high warning	Displays whether the module temperature high warning is **On** or **Off**.
Module temperature low warning	Displays whether the module temperature low warning is **On** or **Off**.
Module voltage high alarm	Displays whether the module voltage high alarm is **On** or **Off**.
Module voltage low alarm	Displays whether the module voltage low alarm is **On** or **Off**.
Module voltage high warning	Displays whether the module voltage high warning is **On** or **Off**.
Module voltage low warning	Displays whether the module voltage low warning is **On** or **Off**.
Laser rx power high alarm	Displays whether the receive laser power high alarm is **On** or **Off**.
Laser rx power low alarm	Displays whether the receive laser power low alarm is **On** or **Off**.
Laser rx power high warning	Displays whether the receive laser power high warning is **On** or **Off**.
Laser rx power low warning	Displays whether the receive laser power low warning is **On** or **Off**.
Laser bias current high alarm threshold	Displays the vendor-specified threshold for the laser bias current high alarm.
Laser bias current low alarm threshold	Displays the vendor-specified threshold for the laser bias current low alarm.
Laser bias current high warning threshold	Displays the vendor-specified threshold for the laser bias current high warning.
Laser bias current low warning threshold	Displays the vendor-specified threshold for the laser bias current low warning.
Laser output power high alarm threshold	Displays the vendor-specified threshold for the laser output power high alarm.
Laser output power low alarm threshold	Displays the vendor-specified threshold for the laser output power low alarm.

Table 36: show interfaces diagnostics optics Output Fields *(continued)*

Field Name	Field Description
Laser output power high warning threshold	Displays the vendor-specified threshold for the laser output power high warning.
Laser output power low warning threshold	Displays the vendor-specified threshold for the laser output power low warning.
Module temperature high alarm threshold	Displays the vendor-specified threshold for the module temperature high alarm.
Module temperature low alarm threshold	Displays the vendor-specified threshold for the module temperature low alarm.
Module temperature high warning threshold	Displays the vendor-specified threshold for the module temperature high warning.
Module temperature low warning threshold	Displays the vendor-specified threshold for the module temperature low warning.
Module voltage high alarm threshold	Displays the vendor-specified threshold for the module voltage high alarm.
Module voltage low alarm threshold	Displays the vendor-specified threshold for the module voltage low alarm.
Module voltage high warning threshold	Displays the vendor-specified threshold for the module voltage high warning.
Module voltage low warning threshold	Displays the vendor-specified threshold for the module voltage low warning.
Laser rx power high alarm threshold	Displays the vendor-specified threshold for the laser rx power high alarm.
Laser rx power low alarm threshold	Displays the vendor-specified threshold for the laser rx power low alarm.
Laser rx power high warning threshold	Displays the vendor-specified threshold for the laser rx power high warning.
Laser rx power low warning threshold	Displays the vendor-specified threshold for the laser rx power low warning.

Sample Output

show interfaces diagnostics optics

user@host> **show interfaces diagnostics optics ge-2/0/0**

```
Physical interface: ge-2/0/0
    Laser bias current                          :  7.408 mA
    Laser output power                          :  0.3500 mW / -4.56 dBm
    Module temperature                          :  23 degrees C / 73 degrees F
    Module voltage                              :  3.3450 V
    Receiver signal average optical power       :  0.0002 mW / -36.99 dBm
    Laser bias current high alarm               :  Off
    Laser bias current low alarm                :  Off
    Laser bias current high warning             :  Off
    Laser bias current low warning              :  Off
    Laser output power high alarm               :  Off
    Laser output power low alarm                :  Off
    Laser output power high warning             :  Off
    Laser output power low warning              :  Off
    Module temperature high alarm               :  Off
    Module temperature low alarm                :  Off
    Module temperature high warning             :  Off
    Module temperature low warning              :  Off
    Module voltage high alarm                   :  Off
    Module voltage low alarm                    :  Off
    Module voltage high warning                 :  Off
    Module voltage low warning                  :  Off
    Laser rx power high alarm                   :  Off
    Laser rx power low alarm                    :  On
    Laser rx power high warning                 :  Off
    Laser rx power low warning                  :  On
    Laser bias current high alarm threshold     :  17.000 mA
    Laser bias current low alarm threshold      :  1.000 mA
    Laser bias current high warning threshold   :  14.000 mA
    Laser bias current low warning threshold    :  2.000 mA
    Laser output power high alarm threshold     :  0.6310 mW / -2.00 dBm
    Laser output power low alarm threshold      :  0.0670 mW / -11.74 dBm
    Laser output power high warning threshold   :  0.6310 mW / -2.00 dBm
    Laser output power low warning threshold    :  0.0790 mW / -11.02 dBm
    Module temperature high alarm threshold     :  95 degrees C / 203 degrees F
    Module temperature low alarm threshold      :  -25 degrees C / -13 degrees F
    Module temperature high warning threshold   :  90 degrees C / 194 degrees F
    Module temperature low warning threshold    :  -20 degrees C / -4 degrees F
    Module voltage high alarm threshold         :  3.900 V
```

```
Module voltage low alarm threshold        :  2.700 V
Module voltage high warning threshold     :  3.700 V
Module voltage low warning threshold      :  2.900 V
Laser rx power high alarm threshold       :  1.2590 mW / 1.00 dBm
Laser rx power low alarm threshold        :  0.0100 mW / -20.00 dBm
Laser rx power high warning threshold     :  0.7940 mW / -1.00 dBm
Laser rx power low warning threshold      :  0.0158 mW / -18.01 dBm
```

show interfaces flow-statistics

Syntax

```
show interfaces flow-statistics   <interface-name>
```

Release Information

Command introduced in Junos OS Release 9.2.

Description

Display interfaces flow statistics.

Options

Interface-name —(Optional) Display flow statistics about the specified interface. Following is a list of typical interface names. Replace *pim* with the PIM slot and *port* with the port number. For a complete list, see the *Interface Naming Conventions*.

- **at-*pim*/0/*port***—ATM-over-ADSL or ATM-over-SHDSL interface.
- **br-*pim*/0/*port***—Basic Rate Interface for establishing ISDN connections.
- **ce1-*pim*/0/*port***—Channelized E1 interface.
- **ct1-*pim*/0/*port***—Channelized T1 interface.
- **dl0**—Dialer Interface for initiating ISDN and USB modem connections.
- **e1-*pim*/0/*port***—E1 interface.
- **e3-*pim*/0/*port***—E3 interface.
- **fe-*pim*/0/ *port***—Fast Ethernet interface.
- **ge-*pim*/0/*port***—Gigabit Ethernet interface.
- **se-*pim*/0/*port***—Serial interface.
- **t1-*pim*/0/*port***—T1 (also called DS1) interface.
- **t3-*pim*/0/ *port***—T3 (also called DS3) interface.
- **wx-*slot*/0/0**—WAN acceleration interface, for the WXC Integrated Services Module (ISM 200).

Required Privilege Level

view

RELATED DOCUMENTATION

List of Sample Output

show interfaces flow-statistics (Gigabit Ethernet) on page 536

Output Fields

Table 37 on page 533 lists the output fields for the **show interfaces flow-statistics** command. Output fields are listed in the approximate order in which they appear.

Table 37: show interfaces flow-statistics Output Fields

Field Name	Field Description
Traffic statistics	Number of packets and bytes transmitted and received on the physical interface.
Local statistics	Number of packets and bytes transmitted and received on the physical interface.
Transit statistics	Number of packets and bytes transiting the physical interface.
Flow input statistics	Statistics on packets received by flow module.
Flow output statistics	Statistics on packets sent by flow module.
Flow error statistics	Packet drop statistics for the flow module. For further details, see Table 38 on page 533.

Table 38: Flow Error Statistics (Packet Drop Statistics for the Flow Module)

Error	Error Description
Screen:	
Address spoofing	The packet was dropped when the screen module detected address spoofing.
Syn-attack protection	The packet was dropped because of SYN attack protection or SYN cookie protection.
VPN:	
Authentication failed	The packet was dropped because the IPsec Encapsulating Security Payload (ESP) or Authentication Header (AH) authentication failed.
No SA for incoming SPI	The packet was dropped because the incoming IPsec packet's security parameter index (SPI) does not match any known SPI.

Table 38: Flow Error Statistics (Packet Drop Statistics for the Flow Module) *(continued)*

Security association not active	The packet was dropped because an IPsec packet was received for an inactive SA.
NAT:	
Incoming NAT errors	The source NAT rule search failed, an invalid source NAT binding was found, or the NAT allocation failed.
Multiple incoming NAT	Sometimes packets are looped through the system more than once; if source NAT is specified more than once, the packet will be dropped.
Auth:	
Multiple user authentications	Sometimes packets are looped through the system more than once. Each time a packet passes through the system, that packet must be permitted by a policy. If the packet matches more than one policy that specifies user authentication, then it will be dropped.
User authentication errors	Packet was dropped because policy requires authentication; however: • Only Telnet, FTP, and HTTP traffic can be authenticated. • The corresponding authentication entry could not be found, if web-auth is specified. • The maximum number of authenticated sessions per user was exceeded.
Flow:	
No one interested in self packets	This counter is incremented for one of the following reasons: • The outbound interface is a self interface, but the packet is not marked as a to-self packet and the destination address is in a source NAT pool. • No service is interested in the to-self packet • When a zone has ident-reset service enabled, the TCP RST to IDENT request for port 113 is sent back and this counter is incremented.
No minor session	The packet was dropped because no minor sessions are available and a minor session was requested. Minor sessions are allocated for storing additional TCP state information.
No more sessions	The packet was dropped because there were no more free sessions available.

Table 38: Flow Error Statistics (Packet Drop Statistics for the Flow Module) *(continued)*

No route present	The packet was dropped because a valid route was not available to forward the packet. For new sessions, the counter is incremented for one of the following reasons: • No valid route was found to forward the packet. • A discard or reject route was found. • The route could not be added due to lack of memory. • The reverse path forwarding check failed for an incoming multicast packet. For existing sessions, the prior route was changed or deleted, or a more specific route was added. The session is rerouted, and this reroute could fail because: • A new route could not be found; either the previous route was removed, or the route was changed to discard or reject. • Multiple packets may concurrently force rerouting to occur, and only one packet can successfully complete the rerouting process. Other packets will be dropped. • The route table was locked for updates by the Routing Engine. Packets that match a new session are retried, whereas packets that match an existing session are not.
No tunnel found	The packet was dropped because a valid tunnel could not be found
No session for a gate	This counter is incremented when a packet is destined for an ALG, and the ALG decides to drop this packet.
No zone or NULL zone binding	The packet was dropped because its incoming interface was not bound to any zone.
Policy denied	The error counter is incremented for one of the following reasons: • Source and/or destination NAT has occurred and policy says to drop the packet. • Policy specifies user authentication, which failed. • Policy was configured to deny this packet.
TCP sequence number out of window	A TCP packet with a sequence number failed the TCP sequence number check that was received.
Counters Not Currently in Use	
No parent for a gate	-
Invalid zone received packet	-
No NAT gate	-

Sample Output

show interfaces flow-statistics (Gigabit Ethernet)

user@host> **show interfaces flow-statistics ge-0/0/1.0**

```
  Logical interface ge-0/0/1.0 (Index 70) (SNMP ifIndex 49)
    Flags: SNMP-Traps Encapsulation: ENET2
    Input packets : 5161
    Output packets: 83
    Security: Zone: zone2
    Allowed host-inbound traffic : bootp bfd bgp  dns dvmrp igmp ldp msdp nhrp
ospf pgm
    pim rip router-discovery rsvp sap vrrp dhcp finger ftp tftp ident-reset http
https ike
   netconf ping rlogin rpm rsh snmp snmp-trap ssh telnet traceroute xnm-clear-text
 xnm-ssl
    lsping
    Flow Statistics :
    Flow Input statistics :
      Self packets :                    0
      ICMP packets :                    0
      VPN packets :                     2564
      Bytes permitted by policy :       3478
      Connections established :         1
    Flow Output statistics:
      Multicast packets :               0
      Bytes permitted by policy :       16994
    Flow error statistics (Packets dropped due to):
      Address spoofing:                 0
      Authentication failed:            0
      Incoming NAT errors:              0
      Invalid zone received packet:     0
      Multiple user authentications:    0
      Multiple incoming NAT:            0
      No parent for a gate:             0
      No one interested in self packets: 0
      No minor session:                 0
      No more sessions:                 0
      No NAT gate:                      0
      No route present:                 0
      No SA for incoming SPI:           0
      No tunnel found:                  0
      No session for a gate:            0
      No zone or NULL zone binding      0
```

```
  Policy denied:                    0
  Security association not active:  0
  TCP sequence number out of window: 0
  Syn-attack protection:            0
  User authentication errors:       0
Protocol inet, MTU: 1500
  Flags: None
  Addresses, Flags: Is-Preferred Is-Primary
    Destination:  203.0.113.1/24, Local: 203.0.113.2, Broadcast: 2.2.2.255
```

show interfaces swfabx

Syntax

```
show interfaces (swfab0 | swfab1)
```

Release Information

Command introduced in Junos OS Release 11.1.

Description

Display the configured interfaces for each swfab interface. The swfab interface can contain one or more members because it is an aggregated interface.

Required Privilege Level

view

RELATED DOCUMENTATION

clear interfaces statistics swfabx

List of Sample Output

show interfaces swfab0 on page 539
show interfaces swfab1 on page 539

Output Fields

Table 39 on page 538 lists the output fields for the **show interfaces <swfab0 | swfab1>** command. Output fields are listed in the approximate order in which they appear.

Table 39: show interfaces <swfab0 | swfab1> Output Fields

Field Name	Field Description
fabric-options	The fabric-options hierarchy is configured to be in sync with the fab interfaces.
member-interfaces	Interfaces specified under member-interfaces are single aggregate interfaces. This interface carries internode switching traffic.

Sample Output

show interfaces swfab0

```
user@host# show interfaces swfab0
```

```
fabric-options {
       member-interfaces {
           ge-0/0/9;
           ge-0/0/10;
       }
  }
```

show interfaces swfab1

```
user@host# show interfaces swfab1
```

```
fabric-options {
       member-interfaces {
           ge-7/0/9;
           ge-7/0/10;
       }
  }
```

show monitor security flow

Syntax

```
show monitor security flow
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10. This topic was updated to include the flow session conn-tag filter in Junos OS Release 15.1X49-D70.

Description

Display information about the security flow session monitoring.

Required Privilege Level

view

RELATED DOCUMENTATION

- Monitoring Security Flow Sessions Overview | 186
- monitor security flow filter | 453
- monitor security flow start | 455
- clear monitor security flow filter | 401

List of Sample Output

show monitor security flow on page 541

Output Fields

Lists the output fields for the **show monitor security flow** command. Output fields are listed in the approximate order in which they appear.

Table 40: show monitor security flow Output Fields

Field Name	Field Description
Monitor security flow session status	State of the security flow session monitoring: **active** or **inactive**.
Monitor security flow trace file	Name of the file for monitoring output.

Table 40: show monitor security flow Output Fields *(continued)*

Field Name	Field Description
Monitor security flow filters	• **conn-tag**—Tag that uniquely identifies a session. The session key is expanded to include this tuple. • **Destination Address**—Address of the destination to be matched. • **Destination Port**—Name of the destination port to be matched. • **Interface Name**—Interface name to be matched. • **Logical System Name**—Logical system name to be matched. • **Name**—Name of the security flow filter. • **Protocol**—Name of the protocol to be matched. • **Source Address**—Address of the source to be matched. • **Source Port**—Name of the source port to be matched. • **Status**—State of the security flow filter: **active** or **inactive**.

Sample Output

show monitor security flow

`user@host>` **show monitor security flow**

```
Monitor security flow session status: Active
Monitor security flow trace file: flow
Monitor security flow filters:
  Name: server-sql
    Status: Active
    source: 10.2.2.1 (port *), destination: 10.20.30.40 (port 1433)
    protocol: TCP
    conn-tag: 0
  Name: internet-access
    Status: Active
    source: * (port *), destination: * (port 80)
    protocol: TCP
    conn-tag: 0
```

show security flow cp-session

Syntax

show security flow cp-session [<filter>] [summary | terse]
<node (***node-id*** | all | local | primary)>

Release Information
Command introduced in Junos OS Release 10.2. Support for connection tag added in Junos OS Release 15.1X49-D40.

Description
Display central point session-related flow information. This command is supported on the SRX5800, SRX5600, and SRX5400 devices.

Options
- **conn**-tag—Session connection tag (0..4294967295)
- **destination-port**—Destination port (1..65535)
- **destination-prefix**—Destination prefix
- **family**—Display session by family.
- **logical-system**—Logical-system name
- **node**—(Optional) For chassis cluster configurations, display security flow cp-session information on a specific node (device) in the cluster.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all** —Display information about all nodes.
  - **local** —Display information about the local node.
  - **primary**—Display information about the primary node.
- **protocol**—IP protocol number
- **root-logical-system**—Root logical-system (default)
- **source-port**—Source port (1..65535)
- **source-prefix**—Source IP prefix or address
- **summary | terse**–Display the specified level of output.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security flow cp-session on page 544
show security flow cp-session summary on page 544
show security flow cp-session terse on page 545

Output Fields

Table 41 on page 543 lists the output fields for the **show security flow cp-session** command. Output fields are listed in the approximate order in which they appear.

Table 41: show security flow cp-session Output Fields

Field Name	Field Description
Valid gates	Number of valid central point sessions.
Pending gates	Number of pending central point sessions.
Invalidated gates	Number of invalid central point sessions.
Gates in other states	Number of central point sessions in other states.
Total gates	Number of central point sessions in total.
Maximum sessions	Number of maximum central point sessions.
Maximum inet6 sessions	Number of maximum inet6 central point sessions.
Session ID	Number that identifies the session. Use this ID to get more information about the session.
Conn Tag	A 32-bit connection tag that uniquely identifies the GPRS tunneling protocol, user plane (GTP-U) and the Stream Control Transmission Protocol (STCP) sessions. The connection tag for GTP-U is the tunnel endpoint identifier (TEID) and for SCTP is the vTag. The connection ID remains 0 if the connection tag is not used by the sessions.
SPU	Services Processing Unit.
In	Incoming flow (source and destination IP addresses).
Out	Reverse flow (source and destination IP addresses).

Sample Output

show security flow cp-session

root> **show security flow cp-session**

```
DCP Flow Sessions on FPC0 PIC0:
Total sessions: 0

DCP Flow Sessions on FPC0 PIC1:

Session ID: 10320276, SPU: 1, Valid
  In: 203.0.113.1/1000 --> o 203.0.113.2/2000;udp, Conn Tag: 0x0,
  Out: 0.0.0.0/0 --> 0.0.0.0/0;0, Conn Tag: 0x0,
Total sessions: 1
```

Sample Output

show security flow cp-session summary

root> **show security flow cp-session summary**

```
DCP Flow Sessions on FPC10 PIC0:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2
Maximum sessions: 7549747
Maximum inet6 sessions: 7549747

DCP Flow Sessions on FPC10 PIC2:
```

```
Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2
Maximum sessions: 7549747
Maximum inet6 sessions: 7549747

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
Maximum sessions: 7549747
Maximum inet6 sessions: 7549747
```

show security flow cp-session terse

root> **show security flow cp-session terse**

```
DCP Flow Sessions on FPC0 PIC1:

Session ID: 10000038, SPU: 1, Valid
  In: 203.0.113.6/1 --> 198.51.100.13/1;pim, Conn Tag: 0x0,
  Out: 198.51.100.13/1 --> 203.0.113.6/1;pim, Conn Tag: 0x0,

Total sessions: 1
```

show security flow cp-session destination-port

Syntax

```
show security flow cp-session destination-port destination-port-number [summary | terse]
```

Release Information

Command introduced in Junos OS Release 10.2. This command is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices.

Description

Display central point session-related flow information for the specified destination port.

Options

- *destination-port-number* —Number of the destination port for which to display central point session information.

  Range: 1 through 65,535

- summary | terse—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow cp-session | 542

show security flow cp-session destination-prefix | 549

List of Sample Output

show security flow cp-session destination-port summary on page 547

show security flow cp-session destination-port terse on page 548

Output Fields

Table 42 on page 546 lists the output fields for the **show security flow cp-session destination-port** command. Output fields are listed in the approximate order in which they appear.

Table 42: show security flow cp-session destination-port Output Fields

Field Name	Field Description
Valid gates	Number of valid central point sessions.

Table 42: show security flow cp-session destination-port Output Fields *(continued)*

Field Name	Field Description
Pending gates	Number of pending central point sessions.
Invalidated gates	Number of invalid central point sessions.
Gates in other states	Number of central point sessions in other states.
Total gates	Number of central point sessions in total.
Session ID	Number that identifies the session. Use this ID to get more information about the session.
SPU	Services Processing Unit.
In	Incoming flow (source and destination IP addresses).
Out	Reverse flow (source and destination IP addresses).

Sample Output

show security flow cp-session destination-port summary

`root>` **show security flow cp-session destination-port 21 summary**

```
DCP Flow Sessions on FPC10 PIC0:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

```
DCP Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow cp-session destination-port terse

root> **show security flow cp-session destination-port 21 terse**

```
DCP Flow Sessions on FPC10 PIC0:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Session ID: 410003298, SPU: 41, Valid
  In: 203.0.113.10/26182 --> 198.51.100.1/21;tcp,
  Out: 198.51.100.1/21 --> 203.0.113.10/26182;tcp,
Total sessions: 1

DCP Flow Sessions on FPC10 PIC2:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow cp-session destination-prefix

Syntax

```
show security flow cp-session destination-prefix destination-IP-prefix  [summary | terse]
```

Release Information

Command introduced in Junos OS Release 10.2. This command is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices.

Description

Display central point session-related flow information for the specified destination prefix.

Options

- *destination-IP-prefix* —Destination IP prefix or address for which to display central point session information.

  Range: 1 through 65,535.

- summary | terse—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow cp-session | 542

show security flow cp-session destination-port | 546

List of Sample Output

show security flow cp-session destination-prefix summary on page 550

show security flow cp-session destination-prefix terse on page 551

Output Fields

Table 43 on page 549 lists the output fields for the **show security flow cp-session destination-prefix** command. Output fields are listed in the approximate order in which they appear.

Table 43: show security flow cp-session destination-prefix Output Fields

Field Name	Field Description
Valid gates	Number of valid central point sessions.

Table 43: show security flow cp-session destination-prefix Output Fields *(continued)*

Field Name	Field Description
Pending gates	Number of pending central point sessions.
Invalidated gates	Number of invalid central point sessions.
Gates in other states	Number of central point sessions in other states.
Total gates	Number of central point sessions in total.
Session ID	Number that identifies the session. Use this ID to get more information about the session.
SPU	Services Processing Unit.
In	Incoming flow (source and destination IP addresses).
Out	Reverse flow (source and destination IP addresses).

Sample Output

show security flow cp-session destination-prefix summary

`root>` **show security flow cp-session destination-prefix 60/8 summary**

```
DCP Flow Sessions on FPC10 PIC0:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

```
DCP Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow cp-session destination-prefix terse

root> **show security flow cp-session destination-prefix 60/8 terse**

```
DCP Flow Sessions on FPC10 PIC0:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:

Session ID: 420002660, SPU: 42, Valid
  In: 203.0.113.10/26183 --> 192.0.2.1/21;tcp,
  Out:192.0.2.1/21 --> 203.0.113.10/26183;tcp,
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow cp-session family

Syntax

```
show security flow cp-session family family [summary | terse]
```

Release Information

Command introduced in Junos OS Release 10.2. This command is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Description

Display central point session-related flow information for the specified family.

Options

- *family*–Display session by family.
- inet—Display IPv4 sessions.
- inet6–Display IPv6 and IPv6-NATPT sessions.
- summary | terse–Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow cp-session | 542

List of Sample Output

show security flow cp-session family summary on page 553
show security flow cp-session family terse on page 554

Output Fields

Table 44 on page 552 lists the output fields for the **show security flow cp-session** family command. Output fields are listed in the approximate order in which they appear.

Table 44: show security flow cp-session family Output Fields

Field Name	Field Description
Valid gates	Number of valid central point sessions.
Pending gates	Number of pending central point sessions.

Table 44: show security flow cp-session family Output Fields *(continued)*

Field Name	Field Description
Invalidated gates	Number of invalid central point sessions.
Gates in other states	Number of central point sessions in other states.
Total gates	Number of central point sessions in total.
Session ID	Number that identifies the session. Use this ID to get more information about the session.
SPU	Services Processing Unit.
In	Incoming flow (source and destination IP addresses).
Out	Reverse flow (source and destination IP addresses).

Sample Output

show security flow cp-session family summary

`root>` **show security flow cp-session family inet summary**

```
DCP Flow Sessions on FPC10 PIC0:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:
```

```
Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow cp-session family terse

root> **show security flow cp-session family inet terse**

```
DCP Flow Sessions on FPC10 PIC0:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:

Session ID: 420002660, SPU: 42, Valid
  In: 198.51.100.1/26183 --> 203.0.113.2/21;tcp,
  Out: 203.0.113.2/21 --> 198.51.100.1/26183;tcp,
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow cp-session protocol

Syntax

```
show security flow cp-session protocol protocol-name  [summary | terse]
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Display central point session-related flow information for the specified protocol.This command is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options

- *protocol-name* —Protocol to use as a central point session filter. Information about the central point session that uses this protocol is displayed.

Possible protocols are:

- ah—IP Security Authentication Header
- egp—Exterior gateway protocol
- esp—IPsec Encapsulating Security Payload
- gre—Generic routing encapsulation
- icmp—Internet Control Message Protocol
- icmp6—Internet Control Message Protocol
- igmp—Internet Group Management Protocol
- ipip—IP over IP
- ospf—Open Shortest Path First
- pim—Protocol Independent Multicast
- rsvp—Resource Reservation Protocol
- sctp—Stream Control Transmission Protocol
- tcp—Transmission Control Protocol
- udp—User Datagram Protocol
- summary | terse–Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow cp-session | 542

List of Sample Output

show security flow cp-session protocol summary on page 556
show security flow cp-session protocol terse on page 557

Output Fields

Table 45 on page 556 lists the output fields for the **show security flow cp-session** protocol command. Output fields are listed in the approximate order in which they appear.

Table 45: show security flow cp-session protocol Output Fields

Field Name	Field Description
Valid gates	Number of valid central point sessions.
Pending gates	Number of pending central point sessions.
Invalidated gates	Number of invalid central point sessions.
Gates in other states	Number of central point sessions in other states.
Total gates	Number of central point sessions in total.
Session ID	Number that identifies the session. Use this ID to get more information about the session.
SPU	Services Processing Unit.
In	Incoming flow (source and destination IP addresses).
Out	Reverse flow (source and destination IP addresses).

Sample Output

show security flow cp-session protocol summary

`root>` **show security flow cp-session protocol tcp summary**

```
DCP Flow Sessions on FPC10 PIC0:
```

```
Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
```

show security flow cp-session protocol terse

`root>` **show security flow cp-session protocol tcp terse**

```
Session ID: 160000015, SPU: 17, Valid
  In: 203.0.113.9/32838 --> 198.51.100.26/21;tcp,
  Out: 198.51.100.26/21 --> 203.0.113.2/32838;tcp,
Total sessions: 1
```

show security flow cp-session source-port

Syntax

```
show security flow cp-session source-port source-port-number [summary | terse]
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Display central point session-related flow information for the specified source-port. This command is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options

source-port-number—Number of the source port about which to display central point session information.

Range: 1 through 65,535

summary | terse—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow cp-session | 542

show security flow cp-session source-prefix | 561

List of Sample Output

show security flow cp-session source-port summary on page 559

show security flow cp-session source-port terse on page 560

Output Fields

Table 46 on page 558 lists the output fields for the **show security flow cp-session source-port** command. Output fields are listed in the approximate order in which they appear.

Table 46: show security flow cp-session source-port Output Fields

Field Name	Field Description
Valid gates	Number of valid central point sessions.
Pending gates	Number of pending central point sessions.

Table 46: show security flow cp-session source-port Output Fields *(continued)*

Field Name	Field Description
Invalidated gates	Number of invalid central point sessions.
Gates in other states	Number of central point sessions in other states.
Total gates	Number of central point sessions in total.
Session ID	Number that identifies the session. Use this ID to get more information about the session.
SPU	Services Processing Unit.
In	Incoming flow (source and destination IP addresses).
Out	Reverse flow (source and destination IP addresses).

Sample Output

show security flow cp-session source-port summary

`root>` **show security flow cp-session source-port 7000 summary**

```
DCP Flow Sessions on FPC10 PIC0:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:
```

```
Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow cp-session source-port terse

root> **show security flow cp-session source-port 7000 terse**

```
DCP Flow Sessions on FPC10 PIC0:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:

Session ID: 420002661, SPU: 42, Valid
  In: 203.0.113.64/7000 --> 192.0.2.9/8000;udp,
  Out: 192.0.2.9/8000 --> 203.0.113.64/7000;udp,
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow cp-session source-prefix

Syntax

```
show security flow cp-session source-prefix source-IP-prefix [summary | terse]
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Display central point session related flow information for the specified source-prefix. This is supported on the SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Options

- *source-IP-prefix*—Source IP prefix or address for which to display central point session information.
- summary | terse—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow cp-session | 542

show security flow cp-session source-port | 558

List of Sample Output

show security flow cp-session source-prefix summary on page 562

show security flow cp-session source-prefix terse on page 563

Output Fields

Table 47 on page 561 lists the output fields for the **show security flow cp-session source-prefix** command. Output fields are listed in the approximate order in which they appear.

Table 47: show security flow cp-session source-prefix Output Fields

Field Name	Field Description
Valid gates	Number of valid central point sessions.
Pending gates	Number of pending central point sessions.
Invalidated gates	Number of invalid central point sessions.

Table 47: show security flow cp-session source-prefix Output Fields *(continued)*

Field Name	Field Description
Gates in other states	Number of central point sessions in other states.
Total gates	Number of central point sessions in total.
Session ID	Number that identifies the session. Use this ID to get more information about the session.
SPU	Services Processing Unit.
In	Incoming flow (source and destination IP addresses).
Out	Reverse flow (source and destination IP addresses).

Sample Output

show security flow cp-session source-prefix summary

`root>` **show security flow cp-session source-prefix 203/8 summary**

```
DCP Flow Sessions on FPC10 PIC0:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessionsons on FPC10 PIC1:

: 0

DCP Flow SessiValid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
```

```
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow cp-session source-prefix terse

root> **show security flow cp-session source-prefix 203/8 terse**

```
DCP Flow Sessions on FPC10 PIC0:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC1:
Total sessions: 0

DCP Flow Sessions on FPC10 PIC2:

Session ID: 420002663, SPU: 42, Valid
  In: 203.0.113.10/7000 --> 198.51.100.2/8000;udp,
  Out: 198.51.100.2/8000 --> 203.0.113.10/7000;udp,
Total sessions: 1

DCP Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow gate

Syntax

```
show security flow gate
[<filter>] [brief | summary}
```

Release Information

Command introduced in Junos OS Release 8.5; Filter and display options added in Junos OS Release 10.2.

Description

Display information about temporary openings known as pinholes or gates in the security firewall.

Pinholes are used by applications that commonly have both control and data sessions and must create openings in the firewall for the data sessions based on information from the parent sessions.

Options

- destination-port—Destination port
- destination-prefix—Destination IP prefix or address
- protocol—IP protocol number
- source-port—Source port
- source-prefix—Source IP prefix or address
- brief | summary—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow gate brief node | 580

show security flow gate destination-port | 588

show security flow gate destination-prefix | 592

show security flow gate protocol | 596

show security flow gate summary node | 600

List of Sample Output

show security flow gate on page 566

show security flow gate brief on page 567

show security flow gate summary on page 568

Output Fields

Table 48 on page 565 lists the output fields for the **show security flow gate** command. Output fields are listed in the approximate order in which they appear.

Table 48: show security flow gate Output Fields

Field Name	Field Description
Hole	Range of flows permitted by the pinhole.
Translated	Tuples used to create the session if it matches the pinhole. • Source address and port • Destination address and port
Protocol	Application protocol, such as UDP or TCP.
Application	Name of the application.
Age	Idle timeout for the pinhole.
Flags	Internal debug flags for the pinhole.
Zone	Incoming zone.
Reference count	Number of resource manager references to the pinhole.
Resource	Resource manager information about the pinhole.
Valid gates	Number of valid gates.
Pending gates	Number of pending gates.
Invalidated gates	Number of invalid gates.
Gates in other states	Number of gates in other states.
Total gates	Number of gates in total.
Maximum gates	Number of maximum gates

Sample Output

show security flow gate

user@host> **show security flow gate**

```
Hole: 0.0.0.0-0.0.0.0/0-0->40.1.1.198.51.100.252/64515-64515
Translated: 0.0.0.0/0->10.0.31.161/25415
Protocol: udp
Application: none/0
Age: 101 seconds
Flags: 0xe001
Zone: untrust
Reference count: 1
Resource: 5-1024-8185
Hole: 0.0.0.0-0.0.0.0/0-198.51.100.252/1046-1046
Translated: 198.51.100.252/36039-> 203.0.113.1/5060
Protocol: udp
Application: junos-sip/63
Age: 65535 seconds
Flags: 0xe200
Zone: untrust
Reference count: 1
Resource: 5-1024-8189
Hole: 0.0.0.0-0.0.0.0/0-0->198.51.100.252-198.51.100.252/24101-24101
Translated: 0.0.0.0/0-> 198.51.100.252/24101
Protocol: udp
Application: none/0
Age: 93 seconds
Flags: 0xe001
Zone: trust
Reference count: 1
Resource: 5-1024-8188
Hole: 0.0.0.0-0.0.0.0/0-0->40.1.1.5-198.51.100.252/24100-24100
Translated: 0.0.0.0/0->198.51.100.252/24100
Protocol: udp
Application: none/0
Age: 93 seconds
Flags: 0xe001
Zone: trust
Reference count: 1
Resource: 5-1024-8191
Hole: 0.0.0.0-0.0.0.0/0-0->198.51.100.252-198.51.100.252/5060-5060
Translated: 0.0.0.0/0->198.51.100.252/5060
Protocol: udp
```

```
Application: junos-sip/63
Age: 65535 seconds
Flags: 0xe200
Zone: trust
Reference count: 1
Resource: 5-1024-8190
```

show security flow gate brief

root> **show security flow gate brief**

```
Flow Gates on FPC4 PIC1:

Hole: 192.0.2.1-192.0.2.1/0->192.0.2.100-192.0.2.100/38143-38143
  Translated: 192.0.2.1->192.0.2.100/38143
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65532 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC5 PIC0:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate summary

root> **show security flow gate summary**

```
Flow Gates on FPC4 PIC1:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC5 PIC0:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

show security flow ip-action

Syntax

```
show security flow ip-action  [ <filter> ] [ summary family (inet | inet6) ]
```

Release Information

Command introduced in Junos OS Release 10.1. Logical systems option added in Junos OS Release 11.2 . Summary option introduced in Junos OS Release 12.1.

Description

Display the current IP-action settings, based on filtered options, for IP sessions running on the device.

Options

- *filter*—Filter the display based on the specified criteria.

  The following filters display those sessions that match the criteria specified by the filter. Refer to the sample output for filtered output examples.

  all | [*filter*]—All active sessions on the device.

  destination-port *destination-port*—Destination port number of the traffic. Range is 1 through 65,535.

  destination-prefix *destination-prefix*—Destination IP prefix or address.

  family (inet | inet6) [*filter*]—IPv4 traffic or IPv6-NATPT traffic and filtered options.

  logical-system *logical-system-name* | all [*filter*]—Specified logical system or all logical systems.

  protocol *protocol-name* | *protocol-number* [*filter*]—Protocol name or number and filtered options.

  - **ah** or **51**
  - **egp** or **8**
  - **esp** or **50**
  - **gre** or **47**
  - **icmp** or **1**
  - **icmp6** or **58**
  - **ipip** or **4**
  - **ospf** or **89**
  - **pim** or **103**
  - **rsvp** or **46**
  - **sctp** or **132**

- **tcp** or **6**
- **udp** or **17**

root-logical-system [*filter*]—Default logical system information and filtered options.

source-port ***source-port***—Source port number of the traffic. Range is 1 through 65,535.

source-prefix ***source-prefix***—Source IP prefix or address of the traffic.

- **summary** —Summary information about IP-action entries.

  family—Display summary of IP-action entries by family. This option is used to filter the output.

  - **inet**—Display summary of IPv4 entries.
  - **inet6**—Display summary of IPv6 entries.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow ip-action | 402

clear security flow session destination-port | 415

List of Sample Output

show security flow ip-action on page 571
show security flow ip-action destination-port on page 572
show security flow ip-action destination-prefix on page 574
show security flow ip-action family inet protocol on page 575
show security flow ip-action family inet logical-system all on page 576
show security flow ip-action source-prefix on page 577
show security flow ip-action summary on page 578
show security flow ip-action summary family inet on page 578
show security flow ip-action summary family inet6 on page 579

Output Fields

Table 49 on page 570 lists the output fields for the **show security flow ip-action** command. Output fields are listed in the approximate order in which they appear.

Table 49: show security flow ip-action Output Fields

Field Name	Field Description
Src-Addr	Source address of outbound IP traffic.

Table 49: show security flow ip-action Output Fields *(continued)*

Field Name	Field Description
Src-Port	Source port number of outbound IP traffic.
Dst-Addr	Destination address of inbound IP traffic.
Dst-Port/Proto	Destination port number and protocol type of inbound IP traffic.
Timeout (sec)	Configured timeouts and time remaining for an IP session.
Zone	Security zone associated with an IP session.
Action	Configured action type, for example, block, close, and notify.
State	The active mode and passive mode describe the states of the **ip-action** entry.
IPv4 action count	The total number of IPv4 entries.
IPv6 action count	The total number of IPv6 entries.

Sample Output

show security flow ip-action

`user@host>` **show security flow ip-action**

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
 203.0.113.1       *       203.0.113.4         21/tcp         293/300      *
     close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *      203.0.113.4          21/tcp         293/300      *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *      203.0.113.4          21/tcp         293/300      *
```

```
 close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *       203.0.113.4          21/tcp         293/300      *
 close        Passive
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *      203.0.113.4          21/tcp         293/300       *
   close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *    203.0.113.4          21/tcp         292/300      *
  close        Passive
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
 203.0.113.1         *    203.0.113.4          21/tcp         292/300      *
  close        Active
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
 IPv6 action count: 0 on FPC0.PIC1
 IPv6 action count: 0 on FPC0.PIC2
 IPv6 action count: 0 on FPC0.PIC3
 IPv6 action count: 0 on FPC1.PIC0
 IPv6 action count: 0 on FPC1.PIC1
 IPv6 action count: 0 on FPC1.PIC2
 IPv6 action count: 0 on FPC1.PIC3
 IPv6 action count: Active mode 0 on all PICs
```

show security flow ip-action destination-port

`user@host>` **show security flow ip-action destination-port 21**

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
```

```
 close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4        21/tcp         274/300      *
 close        Passive
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         274/300      *
 close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1        *        203.0.113.4         21/tcp         274/300      *
 close        Passive
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         273/300      *
 close        Active
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
 IPv6 action count: 0 on FPC0.PIC1
 IPv6 action count: 0 on FPC0.PIC2
 IPv6 action count: 0 on FPC0.PIC3
 IPv6 action count: 0 on FPC1.PIC0
 IPv6 action count: 0 on FPC1.PIC1
```

```
 IPv6 action count: 0 on FPC1.PIC2
 IPv6 action count: 0 on FPC1.PIC3
 IPv6 action count: Active mode 0 on all PICs
```

show security flow ip-action destination-prefix

`user@host>` **show security flow ip-action destination-prefix 203.0.113.4/8**

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1        *        203.0.113.4         21/tcp         245/300      *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
192.0.2.3          *        203.0.113.4         21/tcp         245/300      *
close        Passive
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
192.0.2.3          *        203.0.113.4         21/tcp         245/300      *
close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1        *        203.0.113.4         21/tcp         245/300      *
 close        Passive
 IPv4 action count: 1 on FPC1.PIC2
```

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         21/tcp         245/300      *
close        Active
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

show security flow ip-action family inet protocol

`user@host>` **show security flow ip-action family inet protocoludp**

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
       close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300     *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
   close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
    close        Active
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
    close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
```

```
203.0.113.1        *        203.0.113.4         69/udp         287/300      *
   close        Passive
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        203.0.113.4         69/udp         287/300      *
   close        Passive
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

show security flow ip-action family inet logical-system all

user@host> **show security flow ip-action family inet logical-system all**

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
       close        Passive           root-logical-system
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
close        Passive           root-logical-system
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *       203.0.113.4         69/udp         267/300      *
 close        Passive           root-logical-system
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
 close        Active            root-logical-system
 IPv4 action count: 1 on FPC1.PIC0

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4         69/udp         267/300      *
 close        Passive           root-logical-system
```

```
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4        69/udp         266/300      *
 close        Passive           root-logical-system
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State             Logical-System
203.0.113.1         *        203.0.113.4        69/udp         266/300      *
 close        Passive           root-logical-system
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

show security flow ip-action source-prefix

`user@host>` **show security flow ip-action source-prefix 192.0.2.3/8**

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
 close        Passive
 IPv4 action count: 1 on FPC0.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
  close        Passive
 IPv4 action count: 1 on FPC0.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4        69/udp         244/300      *
   close        Passive
 IPv4 action count: 1 on FPC0.PIC3

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
    close        Active
 IPv4 action count: 1 on FPC1.PIC0
```

```
Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
   close        Passive
 IPv4 action count: 1 on FPC1.PIC1

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
    close        Passiveo
 IPv4 action count: 1 on FPC1.PIC2

Src-Addr        Src-Port Dst-Addr        Dst-Port/Proto Timeout(sec) Zone
Action       State
203.0.113.1         *        192.0.2.4         69/udp         244/300      *
    close        Passive
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
```

show security flow ip-action summary

`user@host>` **show security flow ip-action summary**

```
 IPv4 action count: 1 on FPC0.PIC1
 IPv4 action count: 1 on FPC0.PIC2
 IPv4 action count: 1 on FPC0.PIC3
 IPv4 action count: 1 on FPC1.PIC0
 IPv4 action count: 1 on FPC1.PIC1
 IPv4 action count: 1 on FPC1.PIC2
 IPv4 action count: 1 on FPC1.PIC3
 IPv4 action count: Active mode 1 on all PICs
 IPv6 action count: 0 on FPC0.PIC1
 IPv6 action count: 0 on FPC0.PIC2
 IPv6 action count: 0 on FPC0.PIC3
 IPv6 action count: 0 on FPC1.PIC0
 IPv6 action count: 0 on FPC1.PIC1
 IPv6 action count: 0 on FPC1.PIC2
 IPv6 action count: 0 on FPC1.PIC3
 IPv6 action count: Active mode 0 on all PICs
```

show security flow ip-action summary family inet

`user@host>` **show security flow ip-action summary inet**

```
IPv4 action count: 1 on FPC0.PIC1
IPv4 action count: 1 on FPC0.PIC2
IPv4 action count: 1 on FPC0.PIC3
IPv4 action count: 1 on FPC1.PIC0
IPv4 action count: 1 on FPC1.PIC1
IPv4 action count: 1 on FPC1.PIC2
IPv4 action count: 1 on FPC1.PIC3
IPv4 action count: Active mode 1 on all PICs
```

show security flow ip-action summary family inet6

```
user@host> show security flow ip-action summary family inet6
```

```
IPv6 action count: 1 on FPC0.PIC1
IPv6 action count: 1 on FPC0.PIC2
IPv6 action count: 1 on FPC0.PIC3
IPv6 action count: 1 on FPC1.PIC0
IPv6 action count: 1 on FPC1.PIC1
IPv6 action count: 1 on FPC1.PIC2
IPv6 action count: 1 on FPC1.PIC3
IPv6 action count: Active mode 1 on all PICs
```

show security flow gate brief node

Syntax

```
show security flow gate brief node (node-id | all | local | primary)
```

Release Information

Command introduced in Junos OS Release 8.5; node options added in Junos OS Release 9.0. Filter options added in Junos OS Release 10.2.

Description

Display information about temporary openings known as pinholes or gates in the security firewall for the specified node options in brief mode.

Options

node—(Optional) For chassis cluster configurations, display gate information on a specific node.

- ***node-id*** —Identification number of the node. It can be 0 or 1.
- **all**—Display information about all nodes.
- **local**—Display information about the local node.
- **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow gate | 564
show security flow gate summary node | 600

List of Sample Output

show security flow gate brief node 0 on page 581
show security flow gate brief node 1 on page 582
show security flow gate brief node all on page 583
show security flow gate brief node local on page 585
show security flow gate brief node primary on page 586

Output Fields

Table 50 on page 581 lists the output fields for the **show security flow gate brief node** command. Output fields are listed in the approximate order in which they appear.

Table 50: show security flow gate brief node Output Fields

Field Name	Field Description
Valid gates	Number of valid gates.
Pending gates	Number of pending gates.
Invalidated gates	Number of invalid gates.
Gates in other states	Number of gates in other states.
Total gates	Number of gates in total.
Hole	Range of flows permitted by the pinhole.
Translated	Tuples used to create the session if it matches the pinhole. • Source address and port • Destination address and port
Protocol	Application protocol, such as UDP or TCP.
Application	Name of the application.
Age	Idle timeout for the pinhole.
Flags	Internal debug flags for the pinhole.
Zone	Incoming zone.
Reference count	Number of resource manager references to the pinhole.
Resource	Resource manager information about the pinhole.

Sample Output

show security flow gate brief node 0

```
root@antbert> show security flow gate brief node 0

node0:
--------------------------------------------------------------------------
```

```
Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC4 PIC0:

Hole: 1.0.0.100-1.0.0.100/0-0->2.0.0.100-2.0.0.100/32707-32707
  Translated: 1.0.0.100/0->2.0.0.100/32707
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65518 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate brief node 1

root@antbert> **show security flow gate brief node 1**

```
node1:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:
```

```
Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC4 PIC0:

Hole: 1.0.0.100-1.0.0.100/0-0->2.0.0.100-2.0.0.100/32707-32707
  Translated: 1.0.0.100/0->2.0.0.100/32707
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65514 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate brief node all

root@antbert> **show security flow gate brief node all**

```
node0:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
```

```
Gates in other states: 0
Total gates: 0

Flow Gates on FPC4 PIC0:

Hole: 1.0.0.100-1.0.0.100/0-0->2.0.0.100-2.0.0.100/32707-32707
  Translated: 1.0.0.100/0->2.0.0.100/32707
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65512 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

node1:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC4 PIC0:

Hole: 1.0.0.100-1.0.0.100/0-0->2.0.0.100-2.0.0.100/32707-32707
  Translated: 1.0.0.100/0->2.0.0.100/32707
```

```
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65510 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate brief node local

`root@antbert>` **show security flow gate brief node local**

```
node0:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC4 PIC0:

Hole: 1.0.0.100-1.0.0.100/0-0->2.0.0.100-2.0.0.100/32707-32707
  Translated: 1.0.0.100/0->2.0.0.100/32707
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65504 seconds
```

```
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate brief node primary

`root@antbert>` **show security flow gate brief node primary**

```
node0:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC4 PIC0:

Hole: 1.0.0.100-1.0.0.100/0-0->2.0.0.100-2.0.0.100/32707-32707
  Translated: 1.0.0.100/0->2.0.0.100/32707
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65500 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
```

```
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate destination-port

Syntax

```
show security flow gate destination-port destination-port-number [brief | summary]
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Display information about temporary openings known as pinholes or gates in the security firewall that for the specified destination port.

> **NOTE:** Destination port filter matches the gate only if the given port falls within the range of ports specified in the gate.

Options

- *destination-port-number*—Number of the destination port for which to display gate information.

  Range: 1 through 65,535

- brief | summary—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow gate | 564

show security flow gate destination-prefix | 592

List of Sample Output

show security flow gate destination-port brief on page 589

show security flow gate destination-port summary on page 590

Output Fields

Table 51 on page 589 lists the output fields for the **show security flow gate destination-port** command. Output fields are listed in the approximate order in which they appear.

Table 51: show security flow gate destination-port Output Fields

Field Name	Field Description
Hole	Range of flows permitted by the pinhole.
Translated	Tuples used to create the session if it matches the pinhole. • Source address and port • Destination address and port
Protocol	Application protocol, such as UDP or TCP.
Application	Name of the application.
Age	Idle timeout for the pinhole.
Flags	Internal debug flags for the pinhole.
Zone	Incoming zone.
Reference count	Number of resource manager references to the pinhole.
Resource	Resource manager information about the pinhole.
Valid gates	Number of valid gates.
Pending gates	Number of pending gates.
Invalidated gates	Number of invalid gates.
Gates in other states	Number of gates in other states.
Total gates	Number of gates in total.
Maximum gates	Number of maximum gates.

Sample Output

show security flow gate destination-port brief

```
root> show security flow gate destination-port 33253 brief
```

```
Flow Gates on FPC4 PIC1:

Hole: 40.0.0.111-40.0.0.111/0-0->30.0.0.100-30.0.0.100/33253-33253
  Translated: 40.0.0.111/0->30.0.0.100/33253
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65526 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24576-86016

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC5 PIC0:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate destination-port summary

`root>` **show security flow gate destination-port 33253 summary**

```
Flow Gates on FPC4 PIC1:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
```

```
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC5 PIC0:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

show security flow gate destination-prefix

Syntax

```
show security flow gate destination-prefix destination-IP-prefix [brief | summary]
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Display information about temporary openings known as pinholes or gates in the security firewall for the specified destination prefix.

> **NOTE:** Destination prefix must match both the starting and ending address in the gate.

Options

- *destination-IP-prefix*—Destination IP prefix or address for which to display gate information.
- brief | summary—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow gate | 564

show security flow gate destination-port | 588

List of Sample Output

show security flow gate destination-prefix brief on page 593

show security flow gate destination-prefix summary on page 594

Output Fields

Table 52 on page 593 lists the output fields for the **show security flow gate destination-prefix** command. Output fields are listed in the approximate order in which they appear.

Table 52: show security flow gate destination-prefix Output Fields

Field Name	Field Description
Hole	Range of flows permitted by the pinhole.
Translated	Tuples used to create the session if it matches the pinhole. • Source address and port • Destination address and port
Protocol	Application protocol, such as UDP or TCP.
Application	Name of the application.
Age	Idle timeout for the pinhole.
Flags	Internal debug flags for the pinhole.
Zone	Incoming zone.
Reference count	Number of resource manager references to the pinhole.
Resource	Resource manager information about the pinhole.
Valid gates	Number of valid gates.
Pending gates	Number of pending gates.
Invalidated gates	Number of invalid gates.
Gates in other states	Number of gates in other states.
Total gates	Number of gates in total.

Sample Output

show security flow gate destination-prefix brief

`root>` **show security flow gate destination-prefix 192.0.2.1 brief**

```
Hole: 203.0.113.1-203.0.113.1/0-0->192.0.2.1-192.0.2.1/37308-37308
  Translated: 203.0.113.1/0->192.0.2.1/37308
```

```
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65456 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24575-86015

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC5 PIC0:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate destination-prefix summary

`root>` **show security flow gate destination-prefix 192.0.2.1 summary**

```
Flow Gates on FPC4 PIC1:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC5 PIC0:
```

```
Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate protocol

Syntax

```
show security flow gate protocol protocol-name [brief | summary]
```

Release Information
Command introduced in Junos OS Release 10.2.

Description
Display information about temporary openings known as pinholes or gates in the security firewall for the specified protocol.

Options
- *protocol-name* —Protocol to use as a gate filter. Information about gates that use this protocol is displayed.

Possible protocols are:

- ah—IP Security Authentication Header
- egp—Exterior gateway protocol
- esp—IPsec Encapsulating Security Payload
- gre—Generic routing encapsulation
- icmp—Internet Control Message Protocol
- icmp6—Internet Control Message Protocol
- igmp—Internet Group Management Protocol
- ipip—IP over IP
- ospf—Open Shortest Path First
- pim—Protocol Independent Multicast
- rsvp—Resource Reservation Protocol
- sctp—Stream Control Transmission Protocol
- tcp—Transmission Control Protocol
- udp—User Datagram Protocol
- brief | summary—Display the specified level of output.

Required Privilege Level
view

RELATED DOCUMENTATION

show security flow gate | 564

List of Sample Output

show security flow gate protocol brief on page 598
show security flow gate protocol summary on page 598

Output Fields

Table 53 on page 597 lists the output fields for the **show security flow gate protocol** command. Output fields are listed in the approximate order in which they appear.

Table 53: show security flow gate protocol Output Fields

Field Name	Field Description
Hole	Range of flows permitted by the pinhole.
Translated	Tuples used to create the session if it matches the pinhole. • Source address and port • Destination address and port
Protocol	Application protocol, such as UDP or TCP.
Application	Name of the application.
Age	Idle timeout for the pinhole.
Flags	Internal debug flags for the pinhole.
Zone	Incoming zone.
Reference count	Number of resource manager references to the pinhole.
Resource	Resource manager information about the pinhole.
Valid gates	Number of valid gates.
Pending gates	Number of pending gates.
Invalidated gates	Number of invalid gates.
Gates in other states	Number of gates in other states.
Total gates	Number of gates in total.

Sample Output

show security flow gate protocol brief

root> **root> show security flow gate protocol tcp brief**

```
Hole: 203.0.113.1-40.0.0.111/0-0->192.0.2.1-192.0.2.1/37308-37308
  Translated: 203.0.113.1/0->30.0.0.100/37308
  Protocol: tcp
  Application: FTP ALG/79
  Age: 65414 seconds
  Flags: 0x0080
  Zone: trust
  Reference count: 1
  Resource: 1-24575-86015

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC5 PIC0:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate protocol summary

root> **show security flow gate protocol tcp summary**

```
Flow Gates on FPC4 PIC1:
```

```
Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1

Flow Gates on FPC5 PIC0:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0

Flow Gates on FPC5 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
```

show security flow gate summary node

Syntax

```
show security flow gate summary node (node-id | all | local | primary)
```

Release Information

Command introduced in Junos OS Release 8.5; node options added in Junos OS Release 9.0. Filter options added in Junos OS Release 10.2.

Description

Display information about temporary openings known as pinholes or gates in the security firewall for the specified node options in summary mode.

Options

node—(Optional) For chassis cluster configurations, display gate information on a specific node.

- ***node-id*** —Identification number of the node. It can be 0 or 1.
- **all**—Display information about all nodes.
- **local**—Display information about the local node.
- **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

show security flow gate

show security flow gate brief node | **580**

List of Sample Output

show security flow gate summary node 0 on page 601
show security flow gate summary node 1 on page 602
show security flow gate summary node all on page 603
show security flow gate summary node local on page 604
show security flow gate summary node primary on page 605

Output Fields

Table 54 on page 601 lists the output fields for the **show security flow gate summary node** command. Output fields are listed in the approximate order in which they appear.

Table 54: show security flow gate summary node Output Fields

Field Name	Field Description
Valid gates	Number of valid gates.
Pending gates	Number of pending gates.
Invalidated gates	Number of invalid gates.
Gates in other states	Number of gates in other states.
Total gates	Number of gates in total.
Hole	Range of flows permitted by the pinhole.
Translated	Tuples used to create the session if it matches the pinhole. • Source address and port • Destination address and port
Protocol	Application protocol, such as UDP or TCP.
Application	Name of the application.
Age	Idle timeout for the pinhole.
Flags	Internal debug flags for the pinhole.
Zone	Incoming zone.
Reference count	Number of resource manager references to the pinhole.
Resource	Resource manager information about the pinhole.

Sample Output

show security flow gate summary node 0

```
root@antbert> show security flow gate summary node 0

node0:
--------------------------------------------------------------------------
```

```
Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072

Flow Gates on FPC4 PIC0:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

show security flow gate summary node 1

root@antbert> **show security flow gate summary node 1**

```
node1:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

```
Flow Gates on FPC4 PIC0:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

show security flow gate summary node all

root@antbert> **show security flow gate summary node all**

```
node0:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072

Flow Gates on FPC4 PIC0:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC4 PIC1:
```

```
Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072

node1:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072

Flow Gates on FPC4 PIC0:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

show security flow gate summary node local

root@antbert> **show security flow gate summary node local**

```
node0:
--------------------------------------------------------------------------
```

```
Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072

Flow Gates on FPC4 PIC0:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

show security flow gate summary node primary

root@antbert> **show security flow gate summary node primary**

```
node0:
--------------------------------------------------------------------------

Flow Gates on FPC3 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

```
Flow Gates on FPC4 PIC0:

Valid gates: 1
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 1
Maximum gates: 131072

Flow Gates on FPC4 PIC1:

Valid gates: 0
Pending gates: 0
Invalidated gates: 0
Gates in other states: 0
Total gates: 0
Maximum gates: 131072
```

show security flow pmi statistics

Syntax

```
show security flow pmi statistics
<node (node-id | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 19.2R1.

Description

The **show security flow statistics** command displays information for individual PIC.

Options

- none—Display the security flow statistics information.
- **node**—(Optional) For chassis cluster configurations, display all security flow statistics on a specific node (device) in the cluster.
  - ***node-id***—Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security flow statistics on page 608

Output Fields

Table 55 on page 608 lists the output fields for the **show security flow pmi statistics** command. Output fields are listed in the approximate order in which they appear.

Table 55: show security flow statistics Output Fields

Field Name	Field Description
PMI received packets	Number of PMI inbound packets received.
PMI transmitted packets	Number of PMI outbound packets transmitted.
PMI regular flow path	Number of PMI packets returned using flow path.
PMI dropped packets	Number of PMI packets dropped
PMI encapsulation bytes	Number of PMI bytes encapsulated.
PMI decapsulation bytes	Number of PMI bytes decapsulated.
PMI encapsulation packets	Number of PMI packets encapsulated.
PMI decapsulation packets	Number of PMI packets decapsulated.

Sample Output

show security flow statistics

`user@host>` **show security flow pmi statistics**

```
show security flow pmi statistics
  Flow power-mode-IPSec statistics:
  PMI statistics of FPC2 PIC0:
    PMI received packets:               0
    PMI transmitted packets:            0
    PMI regular flow path:              0
    PMI dropped packets:                0
    PMI encapsulation bytes:            0
    PMI decapsulation bytes:            0
    PMI encapsulation packets:          0
    PMI decapsulation packets:          0

  PMI statistics of FPC2 PIC1:
    PMI received packets:               0
    PMI transmitted packets:            0
    PMI regular flow path:              0
    PMI dropped packets:                0
```

```
    PMI encapsulation bytes:            0
    PMI decapsulation bytes:            0
    PMI encapsulation packets:          0
    PMI decapsulation packets:          0

  PMI statistics summary:
    PMI received packets:               0
    PMI transmitted packets:            0
    PMI regular flow path:              0
    PMI dropped packets:                0
    PMI encapsulation bytes:            0
    PMI decapsulation bytes:            0
    PMI encapsulation packets:          0
    PMI decapsulation packets:          0
```

show security flow session

Syntax

show security flow session [<filter>] [brief | extensive | summary]
<node (***node-id*** | all | local | primary)>

Release Information

Command introduced in Junos OS Release 8.5. Support for filter and view options added in Junos OS Release 10.2.
Application firewall, dynamic application, and logical system filters added in Junos OS Release 11.2.
Policy ID filter added in Junos OS Release 12.3X48-D10.
Support for connection tag added in Junos OS Release 15.1X49-D40.
The **tenant** option introduced in Junos OS Release 18.3R1.

Description

Display information about all currently active security sessions on the device.

> **NOTE:** For the normal flow sessions, the **show security flow session** command displays byte counters based on IP header length. However, for sessions in Express Path mode, the statistics are collected from the IOC2 (SRX5K-MPC), IOC3 (SRX5K-MPC3-100G10G and SRX5K-MPC3-40G10G), and IOC4 (SRX5K-IOC4-MRAT and SRX5K-IOC4-10G) ASIC hardware engines and include full packet length with L2 headers. Because of this, the output displays slightly larger byte counters for sessions in Express Path mode than for the normal flow session.

Options

- *filter*—Filter the display by the specified criteria.

  The following filters reduce the display to those sessions that match the criteria specified by the filter. Refer to the specific **show** command for examples of the filtered output.

  advanced-anti-malware—Show advanced-anti-malware sessions. For details on the **advanced-anti-malware** option, see the Sky Advanced Threat Prevention CLI Reference Guide.

  all-logical-systems-tenants—All multitenancy systems.

  application—Predefined application name.

  application-firewall—Application firewall enabled.

  application-firewall-rule-set—Application firewall enabled with the specified rule set.

  application-traffic-control—Application traffic control session.

application-traffic-control-rule-set—Application traffic control rule set name and rule name.

conn-tag—Session connection tag (0..4294967295).

destination-port—Destination port.

destination-prefix—Destination IP prefix or address.

dynamic-application—Dynamic application.

dynamic-application-group—Dynamic application.

encrypted—Encrypted traffic.

family—Display session by family.

idp—IDP-enabled sessions.

interface—Name of incoming or outgoing interface.

logical-system (all | *logical-system-name*)—Name of a specific logical system or **all** to display all logical systems.

nat—Display sessions with network address translation.

node—(Optional) For chassis cluster configurations, display security flow session information on a specific node (device) in the cluster.

- ***node-id*** —Identification number of the node. It can be 0 or 1.
- **all** —Display information about all nodes.
- **local** —Display information about the local node.
- **primary**—Display information about the primary node.

policy-id—Display session information based on policy ID; the range is 1 through 4,294,967,295.

protocol—IP protocol number.

resource-manager—Resource manager.

root-logical-system—Display root logical system as default.

security-intelligence—Display security intelligence sessions.

services-offload—Display services offload sessions.

session-identifier—Display session with specified session identifier.

source-port—Source port.

source-prefix—Source IP prefix.

tenant—Displays the security flow session information for a tenant system.

tunnel—Tunnel sessions.

- brief | extensive | summary—Display the specified level of output.
- none—Display information about all active sessions.

Required Privilege Level
view

RELATED DOCUMENTATION

> Understanding Traffic Processing on Security Devices | 25
>
> clear security flow session all | 405

List of Sample Output
show security flow session on page 615
show security flow session (with default policy) on page 615
show security flow session brief on page 616
show security flow session extensive on page 616
show security flow session extensive on page 617
show security flow session summary on page 618

Output Fields
Table 56 on page 612 lists the output fields for the **show security flow session** command. Output fields are listed in the approximate order in which they appear.

Table 56: show security flow session Output Fields

Field Name	Field Description	Level of Output
Session ID	Number that identifies the session. Use this ID to get more information about the session.	brief extensive none
If	Interface name.	brief none

Table 56: show security flow session Output Fields *(continued)*

Field Name	Field Description	Level of Output
State	Status of security flow session.	brief extensive none
Conn Tag	A 32-bit connection tag that uniquely identifies the GPRS tunneling protocol, user plane (GTP-U) and the Stream Control Transmission Protocol (STCP) sessions. The connection tag for GTP-U is the tunnel endpoint identifier (TEID) and for SCTP is the vTag. The connection ID remains 0 if the connection tag is not used by the sessions.	brief extensive none
CP Session ID	Number that identifies the central point session. Use this ID to get more information about the central point session.	brief extensive none
Policy name	Name and ID of the policy that the first packet of the session matched.	brief extensive none
Timeout	Idle timeout after which the session expires.	brief extensive none
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).	brief extensive none
Bytes	Number of received and transmitted bytes.	brief extensive none

Table 56: show security flow session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Pkts	Number of received and transmitted packets.	brief extensive none
Total sessions	Total number of sessions.	brief extensive none
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).	brief extensive none
Status	Session status.	extensive
Flag	Internal flag depicting the state of the session, used for debugging purposes.	extensive
Source NAT pool	The name of the source pool where NAT is used.	extensive
Dynamic application	Name of the application.	extensive
Application traffic control rule-set	AppQoS rule set for this session.	extensive
Rule	AppQoS rule for this session.	extensive
Maximum timeout	Maximum session timeout.	extensive
Current timeout	Remaining time for the session unless traffic exists in the session.	extensive
Session State	Session state.	extensive
Start time	Time when the session was created, offset from the system start time.	extensive
Unicast-sessions	Number of unicast sessions.	Summary

Table 56: show security flow session Output Fields *(continued)*

Field Name	Field Description	Level of Output
Multicast-sessions	Number of multicast sessions.	Summary
Services-offload-sessions	Number of services-offload sessions.	Summary
Failed-sessions	Number of failed sessions.	Summary
Sessions-in-use	Number of sessions in use. • Valid sessions • Pending sessions • Invalidated sessions • Sessions in other states	Summary
Maximum-sessions	Maximum number of sessions permitted.	Summary

Sample Output

show security flow session

`root>` **show security flow session**

```
Flow Sessions on FPC0 PIC1:

Session ID: 10115977, Policy name: SG/4, State: Active, Timeout: 56, Valid
  In: 203.0.113.1/1000 --> 203.0.113.11/2000;udp, Conn Tag: 0x0, If: reth1.0, Pkts:
 1, Bytes: 86, CP Session ID: 10320276
  Out: 203.0.113.11/2000 --> 203.0.113.1/1000;udp, Conn Tag: 0x0, If: reth0.0,
Pkts: 0, Bytes: 0, CP Session ID: 10320276

Total sessions: 1
```

show security flow session (with default policy)

`root>` **show security flow session**

```
Session ID: 36, Policy name: pre-id-default-policy/n, Timeout: 2, Valid
  In: 10.10.10.2/61606 --> 10.10.10.1/179;tcp, Conn Tag: 0x0, If: ge-0/0/2.0, Pkts:
 1, Bytes: 64,
```

```
  Out: 10.10.10.1/179 --> 10.10.10.2/61606;tcp, Conn Tag: 0x0, If: .local..0, Pkts:
 1, Bytes: 40,
```

show security flow session brief

```
root> show security flow session brief
```

```
Flow Sessions on FPC0 PIC1:

Session ID: 10115977, Policy name: SG/4, State: Active, Timeout: 62, Valid
  In: 203.0.113.11/1000 --> 203.0.113.1/2000;udp, Conn Tag: 0x0, If: reth1.0, Pkts:
 1, Bytes: 86, CP Session ID: 10320276
  Out: 203.0.113.1/2000 --> 203.0.113.11/1000;udp, Conn Tag: 0x0, If: reth0.0,
Pkts: 0, Bytes: 0, CP Session ID: 10320276

Total sessions: 1
```

show security flow session extensive

```
root> show security flow session extensive
```

```
Flow Sessions on FPC0 PIC1:

Session ID: 10115977, Status: Normal, State: Active
Flags: 0x8000040/0x18000000/0x12000003
Policy name: SG/4
Source NAT pool: Null, Application: junos-gprs-gtp-v0-udp/76
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 90, Current timeout: 54
Session State: Valid
Start time: 6704, Duration: 35
   In: 203.0.113.11/1000 --> 201.11.0.100/2000;udp,
    Conn Tag: 0x0, Interface: reth1.0,
    Session token: 0x6, Flag: 0x40000021
    Route: 0x86053c2, Gateway: 201.10.0.100, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 86
    CP Session ID: 10320276
   Out: 203.0.113.1/2000 --> 203.0.113.11/1000;udp,
    Conn Tag: 0x0, Interface: reth0.0,
```

```
    Session token: 0x7, Flag: 0x50000000
    Route: 0x86143c2, Gateway: 203.0.113.11, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 10320276
Total sessions: 1
```

show security flow session extensive

root> **show security flow session extensive**

```
Flow Sessions on FPC0 PIC0:

Session ID: 10000059, Status: Normal
Flags: 0x10000/0x0/0x10/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: N/A, Current timeout: N/A
Session State: Valid
Start time: 642, Duration: 369
   In: 3.0.0.2/64387 --> 2.0.0.1/8940;esp,
  Conn Tag: 0x0, Interface: xe-2/0/2.0,
    Session token: 0x7, Flag: 0x80100621
    Route: 0xc0010, Gateway: 2.0.0.2, Tunnel: 0
    ESP/AH frag Rx: 0, Generated: 0
    Inner IPv4 frag Rx: 0, Tx: 0, Generated: 0,
    Inner IPv6 frag Rx: 0, Tx: 0, Generated: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 25, Bytes: 3760
    CP Session ID: 0

Session ID: 10000060, Status: Normal
Flags: 0x10000/0x0/0x10/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: N/A, Current timeout: N/A
```

```
Session State: Valid
Start time: 642, Duration: 369
   In: 3.0.0.2/0 --> 2.0.0.1/0;esp,
  Conn Tag: 0x0, Interface: xe-2/0/2.0,
    Session token: 0x7, Flag: 0x621
    Route: 0xc0010, Gateway: 2.0.0.2, Tunnel: 0
    ESP/AH frag Rx: 0, Generated: 0
    Inner IPv4 frag Rx: 0, Tx: 0, Generated: 0,
    Inner IPv6 frag Rx: 0, Tx: 0, Generated: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 0
Total sessions: 2
```

show security flow session summary

`root>` **show security flow session summary**

```
Flow Sessions on FPC10 PIC1:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC10 PIC2:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC10 PIC3:
```

```
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

show security flow session brief node

Syntax

```
show security flow session brief node (node-id | all | local | primary)
```

Release Information

Command introduced in Junos OS Release 8.5; node options added in Junos OS Release 9.0. Filter options added in Junos OS Release 10.2.

Description

Display information about all currently active security sessions on the device for the specified node options in brief mode.

Options

node—(Optional) For chassis cluster configurations, display session information on a specific node.

- ***node-id*** —Identification number of the node. It can be 0 or 1.
- **all**—Display information about all nodes.
- **local**—Display information about the local node.
- **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

> Understanding Traffic Processing on Security Devices | 25
>
> *show security flow session*

List of Sample Output

show security flow session brief node 0 on page 621
show security flow session brief node 1 on page 622
show security flow session brief node all on page 622
show security flow session brief node local on page 623
show security flow session brief node primary on page 624

Output Fields

Table 57 on page 621 lists the output fields for the **show security flow session brief node** command. Output fields are listed in the approximate order in which they appear.

Table 57: show security flow session brief node Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. Use this ID to get more information about the session.
Policy name	Policy that permitted the traffic.
State	Session state.
Timeout	Idle timeout after which the session expires.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
CP Session ID	Number that identifies the central point session. Use this ID to get more information about the central point session.
Total sessions	Total number of sessions.

Sample Output

show security flow session brief node 0

```
root@host> show security flow session brief node 0
```

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000001, Policy name: default-policy-00/2, State: Active, Timeout:
1696, Valid
Resource information : FTP ALG, 1, 0
  In: 203.0.113.1/60059 --> 203.0.113.2/21;tcp, If: reth0.0, Pkts: 14, Bytes: 626,
 CP Session ID: 10000001
  Out: 203.0.113.2/21 --> 203.0.113.1/60059;tcp, If: reth1.0, Pkts: 13, Bytes:
```

```
744, CP Session ID: 10000001

Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0


Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session brief node 1

root@host> **show security flow session brief node 1**

```
node1:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000001, Policy name: default-policy-00/2, State: Active, Timeout:
1696, Valid
Resource information : FTP ALG, 1, 0
  In: 203.0.113.1/60059 --> 203.0.113.2/21;tcp, If: reth0.0, Pkts: 14, Bytes: 626,
 CP Session ID: 10000001
  Out: 203.0.113.2/21 --> 203.0.113.1/60059;tcp, If: reth1.0, Pkts: 13, Bytes:
744, CP Session ID: 10000001
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session brief node all

root@host> **show security flow session brief node all**

```
node0:
--------------------------------------------------------------------------

Session ID: 10000001, Policy name: default-policy-00/2, State: Active, Timeout:
```

```
1696, Valid
Resource information : FTP ALG, 1, 0
  In: 203.0.113.1/60059 --> 203.0.113.2/21;tcp, If: reth0.0, Pkts: 14, Bytes: 626,
 CP Session ID: 10000001
  Out: 203.0.113.2/21 --> 203.0.113.1/60059;tcp, If: reth1.0, Pkts: 13, Bytes:
744, CP Session ID: 10000001
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0

node1:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000001, Policy name: default-policy-00/2, State: Active, Timeout:
1696, Valid
Resource information : FTP ALG, 1, 0
  In: 203.0.113.1/60059 --> 203.0.113.2/21;tcp, If: reth0.0, Pkts: 14, Bytes: 626,
 CP Session ID: 10000001
  Out: 203.0.113.2/21 --> 203.0.113.1/60059;tcp, If: reth1.0, Pkts: 13, Bytes:
744, CP Session ID: 10000001
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session brief node local

`root@host>` **show security flow session brief node local**

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000001, Policy name: default-policy-00/2, State: Active, Timeout:
```

```
1696, Valid
Resource information : FTP ALG, 1, 0
  In: 203.0.113.1/60059 --> 203.0.113.2/21;tcp, If: reth0.0, Pkts: 14, Bytes: 626,
 CP Session ID: 10000001
  Out: 203.0.113.2/21 --> 203.0.113.1/60059;tcp, If: reth1.0, Pkts: 13, Bytes:
744, CP Session ID: 10000001

Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session brief node primary

root@host> **show security flow session brief node primary**

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000001, Policy name: default-policy-00/2, State: Active, Timeout:
1696, Valid
Resource information : FTP ALG, 1, 0
  In: 203.0.113.1/60059 --> 203.0.113.2/21;tcp, If: reth0.0, Pkts: 14, Bytes: 626,
 CP Session ID: 10000001
  Out: 203.0.113.2/21 --> 203.0.113.1/60059;tcp, If: reth1.0, Pkts: 13, Bytes:
744, CP Session ID: 10000001

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session destination-port

Syntax

```
show security flow session destination-port
destination-port-number [brief | extensive | summary]
```

Release Information

Command introduced in Junos OS Release 8.5; Filter and view options added in Junos OS Release 10.2.

Description

Display information about each session that uses the specified destination port.

Options

- ***destination-port-number*** —Number of the destination port for which to display sessions information.
- **Range:** 1 through 65,535

brief | extensive | summary—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow session destination-port | 415

List of Sample Output

show security flow session destination-port 23 on page 627
show security flow session destination-port 23 brief on page 627
show security flow session destination-port 23 extensive on page 628
show security flow session destination-port 23 summary on page 629

Output Fields

Table 58 on page 626 lists the output fields for the **show security flow session destination-port** command. Output fields are listed in the approximate order in which they appear.

Table 58: show security flow session destination-port Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Unicast-sessions	Number of unicast sessions.
Multicast-sessions	Number of multicast sessions.
Failed-sessions	Number of failed sessions.

Table 58: show security flow session destination-port Output Fields *(continued)*

Field Name	Field Description
Sessions-in-use	Number of sessions in use. • Valid sessions • Pending sessions • Invalidated sessions • Sessions in other states
Maximum-sessions	Number of maximum sessions.

Sample Output

show security flow session destination-port 23

`root>` **show security flow session destination-port 23**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000098, Policy name: default-policy-00/2, Timeout: 1778, Valid
  In: 198.51.100.10/15190 --> 198.51.100.2/23;tcp, If: ge-7/1/0.0, Pkts: 109,
Bytes: 5874, CP Session ID: 430000093
  Out: 198.51.100.2/23 --> 198.51.100.10/15190;tcp, If: ge-7/1/1.0, Pkts: 64,
Bytes: 4015, CP Session ID: 430000093
Total sessions: 1
```

show security flow session destination-port 23 brief

`root>` **show security flow session destination-port 23 brief**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
```

```
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000098, Policy name: default-policy-00/2, Timeout: 1778, Valid
  In: 198.51.100.10/15190 --> 198.51.100.2/23;tcp, If: ge-7/1/0.0, Pkts: 109,
Bytes: 5874, CP Session ID: 430000093
  Out: 198.51.100.2/23 --> 198.51.100.10/15190;tcp, If: ge-7/1/1.0, Pkts: 64,
Bytes: 4015, CP Session ID: 430000093
Total sessions: 1
```

show security flow session destination-port 23 extensive

root> **show security flow session destination-port 23 extensive**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000098, Status: Normal
Flags: 0x40/0x0/0x2008003
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-telnet/10
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1630
Session State: Valid
Start time: 65490, Duration: 207
   In: 198.51.100.10/15190 --> 198.51.100.2/23;tcp
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0001021
    Route: 0xa0010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 109, Bytes: 5874
    CP Session ID: 430000093
   Out: 198.51.100.2/23 --> 2198.51.100.10/15190;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0xc0001020
```

```
    Route: 0x80010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 64, Bytes: 4015
    CP Session ID: 430000093
Total sessions: 1
```

show security flow session destination-port 23 summary

root> **show security flow session destination-port 23 summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
```

show security flow session destination-prefix

Syntax

```
show security flow session destination-prefix
destination-IP-prefix [brief | extensive | summary]
```

Release Information

Command introduced in Junos OS Release 8.5. Support for IPv6 addresses in active/active chassis cluster configurations (in addition to the existing support of active/passive chassis cluster configurations) added in Junos OS Release 10.4.
Support for IPv6 addresses added in Junos OS Release 10.2.
Filter and view options added in Junos OS Release 10.2.

Description

Display information about each session that matches the specified IP destination prefix.

Options

- ***destination-IP-prefix***—Destination IP prefix or address for which to display session information.
- brief | extensive | summary–Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
clear security flow session destination-port | 415

List of Sample Output

show security flow session destination-prefix 60/8 on page 632
show security flow session destination-prefix 60/8 brief on page 632
show security flow session destination-prefix 60/8 extensive on page 633
show security flow session destination-prefix 60/8 summary on page 634
show security flow session destination-prefix 10::10 on page 634

Output Fields

Table 59 on page 631 lists the output fields for the **show security flow session destination-prefix** command. Output fields are listed in the approximate order in which they appear.

Table 59: show security flow session destination-prefix Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Unicast-sessions	Number of unicast sessions.
Multicast-sessions	Number of multicast sessions.
Failed-sessions	Number of failed sessions.

Table 59: show security flow session destination-prefix Output Fields *(continued)*

Field Name	Field Description
Sessions-in-use	Number of sessions in use. • Valid sessions • Pending sessions • Invalidated sessions • Sessions in other states
Maximum-sessions	Number of maximum sessions.

Sample Output

show security flow session destination-prefix 60/8

`root>` **show security flow session destination-prefix 60/8**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000098, Policy name: default-policy-00/2, Timeout: 1450, Valid
  In: 192.0.2.10/15190 --> 198.51.100.1/23;tcp, If: ge-7/1/0.0, Pkts: 109, Bytes:
 5874, CP Session ID: 430000093
  Out: 198.51.100.1/23 --> 192.0.2.10/15190;tcp, If: ge-7/1/1.0, Pkts: 64, Bytes:
 4015, CP Session ID: 430000093
Total sessions: 1
```

show security flow session destination-prefix 60/8 brief

`root>` **show security flow session destination-prefix 60/8 brief**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
```

```
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000098, Policy name: default-policy-00/2, Timeout: 1450, Valid
  In: 192.0.2.10/15190 --> 198.51.100.1/23;tcp, If: ge-7/1/0.0, Pkts: 109, Bytes:
 5874, CP Session ID: 430000093
  Out: 198.51.100.1/23 --> 192.0.2.10/15190;tcp, If: ge-7/1/1.0, Pkts: 64, Bytes:
 4015, CP Session ID: 430000093
Total sessions: 1
```

show security flow session destination-prefix 60/8 extensive

root> **show security flow session destination-prefix 60/8 extensive**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000098, Status: Normal
Flags: 0x40/0x0/0x2008003
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-telnet/10
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1172
Session State: Valid
Start time: 65490, Duration: 666
   In: 192.0.2.10/15190 --> 198.51.100.1/23;tcp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0001021
    Route: 0xa0010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 109, Bytes: 5874
    CP Session ID: 430000093
   Out: 198.51.100.1/23 --> 200.0.0.10/15190;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0xc0001020
```

```
    Route: 0x80010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 64, Bytes: 4015
    CP Session ID: 430000093
Total sessions: 1
```

show security flow session destination-prefix 60/8 summary

root> **show security flow session destination-prefix 60/8 summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
```

show security flow session destination-prefix 10::10

user@host> **show security flow session destination-prefix 5001::2**

```
Session ID: 50000004, Policy name: self-traffic-policy/1, Timeout: 2
In: 10::11/42756 --> 10::10/0;icmp, If: .local..0
Out: 10::10/0 --> 10::11/42756;icmp, If: ge-0/3/0.0
```

```
Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
```

show security flow session extensive node

Syntax

```
show security flow session extensive node (node-id | all | local | primary)
```

Release Information

Command introduced in Junos OS Release 8.5; node options added in Junos OS Release 9.0. Filter options added in Junos OS Release 10.2.

Description

Display information about all currently active security sessions on the device for the specified node options in extensive mode.

Options

node—(Optional) For chassis cluster configurations, display session information on a specific node.

- ***node-id*** —Identification number of the node. It can be 0 or 1.
- **all**—Display information about all nodes.
- **local**—Display information about the local node.
- **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

show security flow session | 610

List of Sample Output

show security flow session extensive node 0 on page 637
show security flow session extensive node 1 on page 638
show security flow session extensive node all on page 639
show security flow session extensive node local on page 641
show security flow session extensive node primary on page 642

Output Fields

Table 60 on page 637 lists the output fields for the **show security flow session extensive node** command. Output fields are listed in the approximate order in which they appear.

Table 60: show security flow session extensive node Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Status	Session status.
State	Session state.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Policy that permitted the traffic.
Source NAT pool	The name of the source pool where NAT is used.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Start time	Time when the session was created, offset from the system start time.
Duration	Length of time for which the session is active.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, flag, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, flag, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
CP Session ID	Number that identifies the central point session. Use this ID to get more information about the central point session.

Sample Output

show security flow session extensive node 0

```
root@host> show security flow session extensive node 0
```

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000003, Status: Normal, State: Active
Flags: 0x8000042/0x8000000/0x110103
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1778
Session State: Valid
Start time: 6466, Duration: 28
   In: 10.0.2.1/52080 --> 203.0.113.1/24;tcp,
    Interface: reth0.0,
    Session token: 0x6, Flag: 0x40002621
    Route: 0x86193c2, Gateway: 100.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 414
    CP Session ID: 10000004
   Out: 203.0.113.1/24 --> 10.0.2.1/52080;tcp,
    Interface: reth1.0,
    Session token: 0x6, Flag: 0x40002620
    Route: 0x86033c2, Gateway: 120.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 420
    CP Session ID: 10000004
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session extensive node 1

root@host> **show security flow session extensive node 1**

```
node1:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000003, Status: Normal, State: Backup
Flags: 0x10000042/0x0/0x10103
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 14324
Session State: Valid
Start time: 6248, Duration: 90
   In: 110.0.2.1/52080 --> 203.0.113.1/24;tcp,
    Interface: reth0.0,
    Session token: 0x6, Flag: 0x60002621
    Route: 0x86193c2, Gateway: 100.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 10000003
   Out: 203.0.113.1/24 --> 10.0.2.1/52080;tcp,
    Interface: reth1.0,
    Session token: 0x6, Flag: 0x60002620
    Route: 0x86033c2, Gateway: 120.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 10000003
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session extensive node all

root@host> **show security flow session extensive node all**

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000003, Status: Normal, State: Active
Flags: 0x8000042/0x8000000/0x110103
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1692
Session State: Valid
Start time: 6466, Duration: 113
   In: 10.0.2.1/52080 --> 203.0.113.1/21;tcp,
    Interface: reth0.0,
    Session token: 0x6, Flag: 0x40002621
    Route: 0x86193c2, Gateway: 100.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 414
    CP Session ID: 10000004
   Out: 203.0.113.1/21 --> 10.0.2.1/52080;tcp,
    Interface: reth1.0,
    Session token: 0x6, Flag: 0x40002620
    Route: 0x86033c2, Gateway: 120.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 420
    CP Session ID: 10000004
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0

node1:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:
```

```
Session ID: 10000003, Status: Normal, State: Backup
Flags: 0x10000042/0x0/0x10103
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 14298
Session State: Valid
Start time: 6248, Duration: 115
   In: 10.0.2.1/52080 --> 203.0.113.1/21;tcp,
    Interface: reth0.0,
    Session token: 0x6, Flag: 0x60002621
    Route: 0x86193c2, Gateway: 100.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 10000003
  Out: 203.0.113.1/21 --> 10.0.2.1/52080;tcp,
    Interface: reth1.0,
    Session token: 0x6, Flag: 0x60002620
    Route: 0x86033c2, Gateway: 120.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 10000003
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session extensive node local

root@host> **show security flow session extensive node local**

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000003, Status: Normal, State: Active
```

```
Flags: 0x8000042/0x8000000/0x110103
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1584
Session State: Valid
Start time: 6466, Duration: 221
   In: 100.0.0.2/52080 --> 120.0.0.2/21;tcp,
    Interface: reth0.0,
    Session token: 0x6, Flag: 0x40002621
    Route: 0x86193c2, Gateway: 100.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 414
    CP Session ID: 10000004
   Out: 120.0.0.2/21 --> 100.0.0.2/52080;tcp,
    Interface: reth1.0,
    Session token: 0x6, Flag: 0x40002620
    Route: 0x86033c2, Gateway: 120.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 420
    CP Session ID: 10000004
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session extensive node primary

root@host> **show security flow session extensive node primary**

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:

Session ID: 10000003, Status: Normal, State: Active
Flags: 0x8000042/0x8000000/0x110103
```

```
Policy name: default-policy-00/2
Source NAT pool: Null, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1554
Session State: Valid
Start time: 6466, Duration: 252
   In: 100.0.0.2/52080 --> 120.0.0.2/21;tcp,
    Interface: reth0.0,
    Session token: 0x6, Flag: 0x40002621
    Route: 0x86193c2, Gateway: 100.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 414
    CP Session ID: 10000004
   Out: 120.0.0.2/21 --> 100.0.0.2/52080;tcp,
    Interface: reth1.0,
    Session token: 0x6, Flag: 0x40002620
    Route: 0x86033c2, Gateway: 120.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 420
    CP Session ID: 10000004
Total sessions: 1

Flow Sessions on FPC0 PIC2:
Total sessions: 0

Flow Sessions on FPC0 PIC3:
Total sessions: 0
```

show security flow session family

Syntax

```
show security flow session family (inet | inet6)
[brief | extensive | summary]
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Display filtered summary of information about existing sessions, including types of sessions, active and failed sessions, and the maximum allowed number of sessions.

Options

- **inet**—Display details summary of IPv4 sessions.
- **inet6**—Display details summary of IPv6 sessions.
- brief | extensive | summary–Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow session family | 419

List of Sample Output

show security flow session family inet on page 646
show security flow session family inet brief on page 647
show security flow session family inet extensive on page 647
show security flow session family inet summary on page 649

Output Fields

Table 61 on page 645 lists the output fields for the **show security flow session family** command. Output fields are listed in the approximate order in which they appear.

Table 61: show security flow session family Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. Use this ID to get more information about the session.
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Unicast-sessions	Number of unicast sessions.
Multicast-sessions	Number of multicast sessions.
Failed-sessions	Number of failed sessions.

Table 61: show security flow session family Output Fields *(continued)*

Field Name	Field Description
Sessions-in-use	Number of sessions in use. • Valid sessions • Pending sessions • Invalidated sessions • Sessions in other states
Maximum-sessions	Number of maximum sessions.

Sample Output

show security flow session family inet

`root>` **show security flow session family inet**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000107, Policy name: default-policy-00/2, Timeout: 4, Valid
  In: 203.0.113.0/3 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 420000202
  Out: 203.0.113.4/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000202
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000115, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 430000110
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000110

Session ID: 430000117, Policy name: default-policy-00/2, Timeout: 4, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 430000111
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
```

```
 CP Session ID: 430000111
Total sessions: 2
```

show security flow session family inet brief

root> **show security flow session family inet brief**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000115, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/3 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 420000206
  Out: 203.0.113.4/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000206

Session ID: 420000117, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 420000207
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 420000207
Total sessions: 2

Flow Sessions on FPC10 PIC3:

Session ID: 430000119, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 203.0.113.0/4 --> 203.0.113.5/24;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 430000112
  Out: 203.0.113.5/24 --> 203.0.113.6/24;icmp, If: .local..0, Pkts: 1, Bytes: 84,
 CP Session ID: 430000112
Total sessions: 1
```

show security flow session family inet extensive

root> **show security flow session family inet extensive**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000111, Status: Normal
Flags: 0x80400040/0x0/0x2800023
Policy name: default-policy-00/2
```

```
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 4, Current timeout: 4
Session State: Valid
Start time: 76455, Duration: 0
   In: 203.0.113.0/24 --> 203.0.113.1/24;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0xa0010, Gateway: 203.0.113.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 410000242
   Out: 203.0.113.1/24 --> 203.0.113.10/4;icmp,
    Interface: .local..0,
    Session token: 0x2, Flag: 0x40000030
    Route: 0xfffb0006, Gateway: 203.0.113.1, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 410000242
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Session ID: 420000123, Status: Normal
Flags: 0x80400040/0x0/0x2800023
Policy name: default-policy-00/2
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 4, Current timeout: 2
Session State: Valid
Start time: 76454, Duration: 2
   In: 203.0.113.10/24 --> 203.0.113.11/24;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0xa0010, Gateway: 20010, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
```

```
    CP Session ID: 420000210
   Out: 203.0.113.11/24 --> 203.0.113.12/24;icmp,
    Interface: .local..0,
    Session token: 0x2, Flag: 0x40000030
    Route: 0xfffb0006, Gateway: 203.0.113.1, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 420000210
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000131, Status: Normal
Flags: 0x80400040/0x0/0x2800023
Policy name: default-policy-00/2
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 4, Current timeout: 4
Session State: Valid
Start time: 76421, Duration: 1
   In: 203.0.113.10/24 --> 203.0.113.11/24;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0xa0010, Gateway: 203.0.113.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 430000118
   Out: 203.0.113.12/24 --> 203.0.113.13/24;icmp,
    Interface: .local..0,
    Session token: 0x2, Flag: 0x40000030
    Route: 0xfffb0006, Gateway: 203.0.113.1, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 430000118
Total sessions: 1
```

show security flow session family inet summary

`root>` **show security flow session family inet summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 2
Sessions in other states: 0
Total sessions: 4

Flow Sessions on FPC10 PIC2:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 2
Sessions in other states: 0
Total sessions: 4

Flow Sessions on FPC10 PIC3:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 2
Sessions in other states: 0
Total sessions: 4
```

show security flow session interface

Syntax

```
show security flow session interface
interface-name [brief | extensive | summary]
```

Release Information

Command introduced in Junos OS Release 8.5; Filter and view options added in Junos OS Release 10.2.

Description

Display information about each session that uses the specified interface. The interface name can be a session's incoming or outgoing interface.

Options

- ***interface-name***—Name of the interface on the device for which to display sessions information.
- brief | extensive | summary–Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow session interface | 424

List of Sample Output

show security flow session interface ge-0/0/2.0 on page 653
show security flow session interface ge-0/0/2.0 brief on page 653
show security flow session interface ge-0/0/2.0 extensive on page 654
show security flow session interface ge-7/1/1.0 summary on page 655

Output Fields

Table 62 on page 651 lists the output fields for the **show security flow session interface** command. Output fields are listed in the approximate order in which they appear.

Table 62: show security flow session interface Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.

Table 62: show security flow session interface Output Fields *(continued)*

Field Name	Field Description
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Unicast-sessions	Number of unicast sessions.
Multicast-sessions	Number of multicast sessions.
Failed-sessions	Number of failed sessions.

Table 62: show security flow session interface Output Fields *(continued)*

Field Name	Field Description
Sessions-in-use	Number of sessions in use. • Valid sessions • Pending sessions • Invalidated sessions • Sessions in other states
Maximum-sessions	Number of maximum sessions.

Sample Output

show security flow session interface ge-0/0/2.0

`root>` **show security flow session interface ge-7/1/1.0**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000146, Policy name: default-policy-00/2, Timeout: 58, Valid
  In: 200.0.0.10/9 --> 60.0.0.2/21562;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 420000247
  Out: 60.0.0.2/21562 --> 200.0.0.10/9;icmp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0,
CP Session ID: 420000247
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000146, Policy name: default-policy-00/2, Timeout: 56, Valid
  In: 200.0.0.10/8 --> 60.0.0.2/21562;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 430000131
  Out: 60.0.0.2/21562 --> 200.0.0.10/8;icmp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0,
CP Session ID: 430000131
Total sessions: 1
```

show security flow session interface ge-0/0/2.0 brief

`root>` **show security flow session interface ge-7/1/1.0 brief**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000137, Policy name: default-policy-00/2, Timeout: 2, Valid
  In: 200.0.0.10/5 --> 60.0.0.2/23354;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 410000269
  Out: 60.0.0.2/23354 --> 200.0.0.10/5;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes: 84,
CP Session ID: 410000269
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Session ID: 420000151, Policy name: default-policy-00/2, Timeout: 54, Valid
  In: 200.0.0.10/1 --> 60.0.0.2/23354;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 420000252
  Out: 60.0.0.2/23354 --> 200.0.0.10/1;icmp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0,
CP Session ID: 420000252
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session interface ge-0/0/2.0 extensive

root> **show security flow session interface ge-7/1/1.0 extensive**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000151, Status: Normal
Flags: 0x40/0x0/0x2000003
Policy name: default-policy-00/2
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 60, Current timeout: 48
Session State: Valid
Start time: 83328, Duration: 12
   In: 200.0.0.10/1 --> 60.0.0.2/23354;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0xa0010, Gateway: 200.0.0.10, Tunnel: 0
```

```
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 420000252
   Out: 60.0.0.2/23354 --> 200.0.0.10/1;icmp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0xc0000020
    Route: 0x80010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 420000252
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session interface ge-7/1/1.0 summary

root> **show security flow session interface ge-7/1/1.0 summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2
```

show security flow session nat

Syntax

```
show security flow session nat [brief | extensive | summary]
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Display sessions with network address translation.

Options

brief | extensive | summary—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

show security flow session | 610

List of Sample Output

show security flow session nat brief on page 658
show security flow session nat extensive on page 658
show security flow session nat summary on page 659

Output Fields

Table 63 on page 656 lists the output fields for the **show security flow session nat** command. Output fields are listed in the approximate order in which they appear.

Table 63: show security flow session nat Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.

Table 63: show security flow session nat Output Fields *(continued)*

Field Name	Field Description
Resource information	Information about the session particular to the resource manager, including the name of the ALG, the group ID. and the resource ID.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Valid sessions	Number of valid sessions.
Pending sessions	Number of pending sessions.
Invalidated sessions	Number of invalidated sessions.

Sample Output

show security flow session nat brief

```
root> show security flow session nat brief
```

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000390, Policy name: default-policy-00/2, Timeout: 1778, Valid
  In: 200.0.0.10/41043 --> 60.0.0.2/21;tcp, If: ge-7/1/0.0, Pkts: 9, Bytes: 414,
CP Session ID: 420001090
  Out: 60.0.0.2/21 --> 60.0.0.1/19473;tcp, If: ge-7/1/1.0, Pkts: 8, Bytes: 479,
CP Session ID: 430000964
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session nat extensive

```
root> show security flow session nat extensive
```

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000390, Status: Normal
Flags: 0x2/0x0/0x2010103
Policy name: default-policy-00/2
Source NAT pool: interface, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1770
Session State: Valid
Start time: 151971, Duration: 55
   In: 200.0.0.10/41043 --> 60.0.0.2/21;tcp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0002621
    Route: 0x70010, Gateway: 200.0.0.10, Tunnel: 0
```

```
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 414
    CP Session ID: 420001090
   Out: 60.0.0.2/21 --> 60.0.0.1/19473;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0xe0002620
    Route: 0x80010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 479
    CP Session ID: 430000964
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session nat summary

root> **show security flow session nat summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow session policy-id

Syntax

```
show security flow session policy-id policy-id-number [brief | extensive | summary ]
```

Release Information

Command introduced in Junos OS Release 12.3X48-D10.

Description

Display information about each session by using policy id of the session.

Options

- ***policy-id-number*** —ID of the policy that the first packet of the session matches with.

  Range: 1through 4294967295

- brief | extensive | summary–Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow session protocol | 432

List of Sample Output

show security flow session policy-id 4 on page 662

show security flow session policy-id 4 extensive on page 662

Output Fields

Table 64 on page 660 lists the output fields for the **show security flow session policy-id** command. Output fields are listed in the approximate order in which they appear.

Table 64: show security flow session policy-id Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic.

Table 64: show security flow session policy-id Output Fields *(continued)*

Field Name	Field Description
Timeout	Idle timeout after which the session expires.
In	For the input flow: • Source and destination addresses and protocol tuple for the input flow. • **Interface**: Input flow interface. • **Session token**: Internal token derived from the virtual routing instance. • **Flag**: Internal debugging flags. • **Route**: Internal next hop of the route to be used by the flow. • **Gateway**: Next-hop gateway of the flow. • **Tunnel**: If the flow is going into a tunnel, the tunnel ID. Otherwise, 0 (zero). • **Port Sequence, FIN sequence, FIN state, Cookie**: Internal TCP state tracking information.
Out	For the reverse flow: • Source and destination addresses, and protocol tuple for the reverse flow. • **Interface**: Reverse flow interface. • **Session token**: Internal token derived from the virtual routing instance. • **Flag**: Internal debugging flags. • **Route**: Internal next hop of the route to be used by the flow. • **Gateway**: Next-hop gateway of the flow. • **Tunnel**: If the flow is going into a tunnel, the tunnel ID. Otherwise, 0 (zero). • **Port Sequence, FIN sequence, FIN state, Cookie**: Internal TCP state tracking information.
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Dynamic application	Name of the application.
Maximum timeout	Maximum session timeout.

Table 64: show security flow session policy-id Output Fields *(continued)*

Field Name	Field Description
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.

Sample Output

show security flow session policy-id 4

```
root> show security flow session policy-id 4
```

```
Flow Sessions on FPC1 PIC0:

Session ID: 20093273, Policy name: p1/4, Timeout: 1784, Valid
  In: 101.0.0.2/1 --> 111.0.0.3/1;0, If: ge-0/0/0.0, Pkts: 1, Bytes: 84
  Out: 111.0.0.3/1 --> 201.0.0.1/22643;0, If: ge-0/0/1.0, Pkts: 0, Bytes: 0
Total sessions: 1
```

show security flow session policy-id 4 extensive

```
root> show security flow session policy-id 4 extensive
```

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000428, Status: Normal
Flags: 0x0/0x0/0x2008003
Policy name: p1/4
Source NAT pool: interface, Application: junos-telnet/10
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1740
Session State: Valid
Start time: 152305, Duration: 64
```

```
   In: 200.0.0.10/15192 --> 60.0.0.2/23;tcp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0001021
    Route: 0x70010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 40, Bytes: 2251
    CP Session ID: 420001128
   Out: 60.0.0.2/23 --> 60.0.0.1/8078;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0xe0001020
    Route: 0x80010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 28, Bytes: 1714
    CP Session ID: 430000965
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session protocol

Syntax

show security flow session protocol (***protocol-name*** | ***protocol-number***)
[brief | extensive | summary]

Release Information
Command introduced in Junos OS Release 8.5; Filter and view options introduced in Junos OS Release 10.2.

Description
Display information about each session that uses the specified protocol.

Options
protocol-name —(Optional) Protocol to use as a sessions filter. Information about sessions that use this protocol is displayed. Possible protocols are:

- **ah**—IP Security Authentication Header
- **egp**—Exterior gateway protocol
- **esp**—IPsec Encapsulating Security Payload
- **gre**—Generic routing encapsulation
- **icmp**—Internet Control Message Protocol
- **igmp**—Internet Group Management Protocol
- **ipip**—IP over IP
- **ospf**—Open Shortest Path First
- **pim**—Protocol Independent Multicast
- **rsvp**—Resource Reservation Protocol
- **sctp**—Stream Control Transmission Protocol
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

protocol-number —(Optional) Numeric protocol value. For a complete list of possible numeric values, see RFC 1700, *Assigned Numbers* (*for the Internet Protocol Suite*).

Range: 0 through 255

brief | extensive | summary—Display the specified level of output.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
clear security flow session protocol | 432

List of Sample Output

show security flow session protocol icmp on page 666
show security flow session protocol icmp brief on page 667
show security flow session protocol icmp extensive on page 668
show security flow session protocol icmp summary on page 669

Output Fields

Table 65 on page 665 lists the output fields for the **show security flow session protocol** command. Output fields are listed in the approximate order in which they appear.

Table 65: show security flow session protocol Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.

Table 65: show security flow session protocol Output Fields *(continued)*

Field Name	Field Description
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Unicast-sessions	Number of unicast sessions.
Multicast-sessions	Number of multicast sessions.
Failed-sessions	Number of failed sessions.
Sessions-in-use	Number of sessions in use. • Valid sessions • Pending sessions • Invalidated sessions • Sessions in other states
Maximum-sessions	Number of maximum sessions.

Sample Output

show security flow session protocol icmp

`root>` **show security flow session protocol icmp**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000654, Policy name: p1/4, Timeout: 2, Valid
  In: 200.0.0.10/2 --> 60.0.0.2/15685;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 410001264
  Out: 60.0.0.2/15685 --> 200.0.0.10/2;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes: 84,
```

```
CP Session ID: 410001264
Total sessions: 1

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000399, Policy name: p1/4, Timeout: 2, Valid
  In: 200.0.0.10/3 --> 60.0.0.2/15685;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 430001053
  Out: 60.0.0.2/15685 --> 200.0.0.10/3;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes: 84,
CP Session ID: 430001053
Total sessions: 1
```

show security flow session protocol icmp brief

`root>` **show security flow session protocol icmp brief**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000658, Policy name: p1/4, Timeout: 4, Valid
  In: 200.0.0.10/4 --> 60.0.0.2/16453;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 410001268
  Out: 60.0.0.2/16453 --> 200.0.0.10/4;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes: 84,
CP Session ID: 410001268
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Session ID: 420000612, Policy name: p1/4, Timeout: 2, Valid
  In: 200.0.0.10/5 --> 60.0.0.2/16453;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 420001316
  Out: 60.0.0.2/16453 --> 200.0.0.10/5;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes: 84,
CP Session ID: 420001316
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Session ID: 430000405, Policy name: p1/4, Timeout: 2, Valid
  In: 200.0.0.10/6 --> 60.0.0.2/16453;icmp, If: ge-7/1/0.0, Pkts: 1, Bytes: 84,
CP Session ID: 430001059
  Out: 60.0.0.2/16453 --> 200.0.0.10/6;icmp, If: ge-7/1/1.0, Pkts: 1, Bytes: 84,
```

```
CP Session ID: 430001059
Total sessions: 1
```

show security flow session protocol icmp extensive

root> **show security flow session protocol icmp extensive**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000660, Status: Normal
Flags: 0x80000040/0x0/0x2800003
Policy name: p1/4
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 4, Current timeout: 2
Session State: Valid
Start time: 153201, Duration: 3
   In: 200.0.0.10/8 --> 60.0.0.2/16453;icmp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0000021
    Route: 0x70010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 410001270
   Out: 60.0.0.2/16453 --> 200.0.0.10/8;icmp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0xc0000020
    Route: 0x80010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 1, Bytes: 84
    CP Session ID: 410001270
Total sessions: 1

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session protocol icmp summary

root> **show security flow session protocol icmp summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 1
Sessions in other states: 0
Total sessions: 3

Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 1
Sessions in other states: 0
Total sessions: 3
```

show security flow session resource-manager

Syntax

show security flow session resource-manager
[brief | extensive | summary]

Release Information
Command introduced in Junos OS Release 8.5; Filter and view options introduced in Junos OS Release 10.2.

Description
Display information about sessions created by the resource manager.

Options
none—Display all resource manager sessions.

brief | extensive | summary—Display the specified level of output.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
clear security flow session resource-manager | 435

List of Sample Output
show security flow session resource-manager on page 672
show security flow session resource-manager brief on page 672
show security flow session resource-manager extensive on page 673
show security flow session resource-manager summary on page 674

Output Fields
Table 66 on page 670 lists the output fields for the **show security flow session resource-manager** command. Output fields are listed in the approximate order in which they appear.

Table 66: show security flow session resource-manager Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.

Table 66: show security flow session resource-manager Output Fields *(continued)*

Field Name	Field Description
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
Resource information	Information about the session particular to the resource manager, including the name of the ALG, the group ID. and the resource ID.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Valid sessions	Number of valid sessions.
Pending sessions	Number of pending sessions.
Invalidated sessions	Number of invalidated sessions.

Table 66: show security flow session resource-manager Output Fields *(continued)*

Field Name	Field Description
Sessions in other states	Number of sessions in other states.
CP Session ID	Number that identifies the central point session. Use this ID to get more information about the central point session.

Sample Output

show security flow session resource-manager

`root>` **show security flow session resource-manager**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000664, Policy name: p1/4, Timeout: 1734, Valid
Resource information : FTP ALG, 1, 0
  In: 200.0.0.10/41047 --> 60.0.0.2/21;tcp, If: ge-7/1/0.0, Pkts: 13, Bytes: 586,
 CP Session ID: 410001274
  Out: 60.0.0.2/21 --> 200.0.0.10/41047;tcp, If: ge-7/1/1.0, Pkts: 13, Bytes: 803,
 CP Session ID: 410001274
Total sessions: 1

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session resource-manager brief

`root>` **show security flow session resource-manager brief**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000664, Policy name: p1/4, Timeout: 1704, Valid
Resource information : FTP ALG, 1, 0
  In: 200.0.0.10/41047 --> 60.0.0.2/21;tcp, If: ge-7/1/0.0, Pkts: 13, Bytes: 586,
 CP Session ID: 410001274
  Out: 60.0.0.2/21 --> 200.0.0.10/41047;tcp, If: ge-7/1/1.0, Pkts: 13, Bytes: 803,
```

```
 CP Session ID: 410001274
Total sessions: 1

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session resource-manager extensive

root> **show security flow session resource-manager extensive**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000664, Status: Normal
Flags: 0x42/0x0/0x2010103
Policy name: p1/4
Source NAT pool: Null, Application: junos-ftp/1
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1682
Session State: Valid
Start time: 160496, Duration: 153
Client: FTP ALG, Group: 1, Resource: 0
   In: 200.0.0.10/41047 --> 60.0.0.2/21;tcp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0xc0002621
    Route: 0x70010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 13, Bytes: 586
    CP Session ID: 410001274
   Out: 60.0.0.2/21 --> 200.0.0.10/41047;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0xc0002620
    Route: 0x80010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 13, Bytes: 803
    CP Session ID: 410001274
Total sessions: 1
```

```
Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session resource-manager summary

root> **show security flow session resource-manager summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow session services-offload

Syntax

show security flow session services-offload
[*filter*] [**brief | extensive | summary**]

Release Information
Command introduced in Junos OS Release 11.4.
Low-latency option introduced in Junos OS Release 12.1X44-D10.
Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the SRX5K-MPC3-100G10G (IOC3) and the SRX5K-MPC3-40G10G (IOC3) with Express Path (formerly known as *services offloading*) support are introduced for SRX5400, SRX5600, and SRX5800 Series devices.
Support added on SRX5800, SRX5600, SRX5400 Series devices and vSRX.
Support added on SRX4600 Series device in Junos OS Release 20.1R1.
Support added at the logical system and tenant system level in Junos OS Release 20.1R1.

Description
Display information about all currently active services-offload security sessions on the device.

Options
- *filter*—Filter the display by the specified criteria.

  The following filters reduce the display to those sessions that match the criteria specified by the filter:

  application —Application name.

  application-firewall-rule-set—Application firewall enabled with the specified rule set.

  application-traffic-control-rule-set—Application traffic control enabled with the specified rule set.

  destination-port—Destination port.

  destination-prefix—Destination IP prefix or address.

  dynamic-application—Dynamic application name.

  dynamic-application-group—Dynamic application group name.

  encrypted—Show encrypted traffic.

  family—Protocol family.

  interface—Name of incoming or outgoing interface.

  logical-system—Logical system name.

  protocol—IP protocol number.

root-logical-system—Root logical system name.

source-port—Source port.

source-prefix—Source IP prefix or address.

tenant—Tenant system name.

- **brief | extensive | summary**—Display the specified level of output.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
clear security flow session services-offload | 437

List of Sample Output

show security flow session services-offload on page 678
show security flow session services-offload brief on page 678
show security flow session services-offload extensive on page 679
show security flow session services-offload summary on page 680
show security flow session services-offload summary logical-system LSYS1 on page 680
show security flow session services-offload summary on page 681
show security flow session services-offload summary tenant TSYS1 on page 681
show security flow session services-offload summary on page 681

Output Fields

Table 67 on page 676 lists the output fields for the **show security flow session services-offload** command. Output fields are listed in the approximate order in which they appear.

Table 67: show security flow session services-offload Output Fields

Field Name	Field Description
Session ID	Number that identifies the services-offload session. Use this ID to get more information about the session.
Policy name	Policy that permits the services-offload traffic.
Timeout	Idle timeout period after which the services-offload session expires.

Table 67: show security flow session services-offload Output Fields *(continued)*

Field Name	Field Description
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets, and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets, and bytes).
Total sessions	Total number of services-offload sessions.
Status	Services-offload session status.
Flag	Internal flag depicting the state of the services-offload session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the services-offload session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Dynamic application	Name of the dynamic application.
Maximum timeout	Maximum amount of idle time allowed for the services-offload session.
Current timeout	Number of seconds that the current services-offload session has been idle.
Session State	Services-offload session state.
Start time	Time when the services-offload session was created, offset from the system start time.
Duration	Duration of the services-offload session.
Valid sessions	Number of valid services-offload sessions.
Pending sessions	Number of pending services-offload sessions.
Invalidated sessions	Number of invalidated services-offload sessions.

Table 67: show security flow session services-offload Output Fields *(continued)*

Field Name	Field Description
Sessions in other states	Number of services-offload sessions in other states.
Total sessions	Total number of services-offload sessions.

Sample Output

show security flow session services-offload

user@host>**show security flow session services-offload**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Policy name: p1/4, Timeout: 1788, Valid
  In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp, If: ge-7/1/0.0, Pkts: 9, Bytes: 507,
CP Session ID: 420000002
  Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp, If: ge-7/1/1.0, Pkts: 8, Bytes: 462,
 CP Session ID: 420000002
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session services-offload brief

user@host>**show security flow session services-offload brief**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Policy name: p1/4, Timeout: 1748, Valid
  In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp, If: ge-7/1/0.0, Pkts: 9, Bytes: 507,
CP Session ID: 420000002
  Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp, If: ge-7/1/1.0, Pkts: 8, Bytes: 462,
```

```
 CP Session ID: 420000002
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session services-offload extensive

user@host>**show security flow session services-offload extensive**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Status: Normal
Flags: 0x40/0x0/0x2408003, services-offload
Policy name: p1/4
Source NAT pool: Null, Application: junos-telnet/10
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 1718
Session State: Valid
Start time: 165, Duration: 89
   In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0x42001021
    Route: 0x80010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 507
    CP Session ID: 420000002
   Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x42001020
    Route: 0x70010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 462
    CP Session ID: 420000002
Total sessions: 1
```

```
Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session services-offload summary

user@host>**show security flow session services-offload summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
```

show security flow session services-offload summary logical-system LSYS1

user@host>**show security flow session services-offload summary logical-system LSYS1**

```
Valid sessions: 500
           Pending sessions: 0
           Invalidated sessions: 0
           Sessions in other states: 0
           Total sessions: 500
```

show security flow session services-offload summary

user@host:LSYS1>**show security flow session services-offload summary**

```
Valid sessions: 500
           Pending sessions: 0
           Invalidated sessions: 0
           Sessions in other states: 0
           Total sessions: 500
```

show security flow session services-offload summary tenant TSYS1

user@host>**show security flow session services-offload summary tenant TSYS1**

```
Valid sessions: 10
           Pending sessions: 0
           Invalidated sessions: 0
           Sessions in other states: 0
           Total sessions: 10
```

show security flow session services-offload summary

user@host:TSYS1>**show security flow session services-offload summary**

```
Valid sessions: 10
           Pending sessions: 0
           Invalidated sessions: 0
           Sessions in other states: 0
           Total sessions: 10
```

show security flow session session-identifier

Syntax

```
show security flow session session-identifier session-identifier
```

Release Information

Command introduced in Junos OS Release 8.5. Output changed to support natflag2 and flag in Junos OS Release 12.3X48-D10.

Description

Display detailed information for the identified session.

Options

session-identifier —Identifier of the session about which to display information.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow session session-identifier | 441

List of Sample Output

show security flow session session-identifier 20595 on page 686

Output Fields

Table 68 on page 682 lists the output fields for the **show security flow session session-identifier** command. Output fields are listed in the approximate order in which they appear.

Table 68: show security flow session session-identifier Output Fields

Field Name	Field Description
Session ID	A unique number that a server assigns a specific user for the duration of that session.

Table 68: show security flow session session-identifier Output Fields *(continued)*

Field Name	Field Description
Flags	Internal flag depicting the state of the session, used for debugging purposes. It is internal. The three available flags are: Flags: 0x42/0x0/0x1/0x8103 • natflag: • natflag2: • natflag3: • flag:
Policy name	Name and ID of the policy that the first packet of the session matched and that permitted the traffic. Session log entries are tied to policy configuration. Each main session event—create, close, and deny—creates a log entry if the controlling policy has enabled logging.
Source NAT pool	The name of the source pool where NAT is used. A NAT pool is a user-defined set of IP addresses that are used for translation. Unlike static NAT, where there is a one-to-one mapping that includes destination IP address translation in one direction and source IP address translation in the reverse direction, with source NAT, the original source IP address is translated to an IP address in the address pool. Source NAT is used to allow hosts with private IP addresses to access a public network.
Dynamic application	Dynamic application: INCONCLUSIVE. If the dynamic application has yet to be determined, the output indicates Pending. If the dynamic application cannot be determined, the output indicates junos: UNKNOWN. Traffic with an application ID of junos: UNKNOWN matches a dynamic application of junos: UNKNOWN. If there is no such rule defined, the default rule is applied. The term junos: UNKNOWN is a reserved keyword.
Encryption	Type of encryption, if the application traffic is encrypted. Encryption: Unknown.
Application traffic control rule-set	Name of the application traffic control rule set. Application traffic control rule-set: INVALID.

Table 68: show security flow session session-identifier Output Fields *(continued)*

Field Name	Field Description
Maximum timeout	Maximum session timeout, in seconds.
Session state	Session state: Valid.
Start time	Time, in seconds, when the session was created, offset from the system start time.
In	For the input flow: • Source and destination addresses, ports, and protocol tuple for the input flow. • **Interface**: Input flow interface. • **Session token**: Internal token derived from the virtual routing instance. • **Flag**: Internal debugging flags. • **Route**: Internal next hop of the route to be used by the flow. • **Gateway**: Next-hop gateway of the flow. • **Tunnel**: Used for internal debugging. If the flow is going into a tunnel, the decimal format of the tunnel ID, plus the tunnel type. Otherwise, 0 (zero). See Table 69 on page 686 for tunnel type identification. • **Port Sequence, FIN sequence, FIN state, Cookie**: Internal TCP state tracking information. • **Pkts, Bytes, CP Session ID**: Packets and bytes matched on the wing, and the associated CP session ID of the wing. • **Conn tag**: Session connection tag for GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.

Table 68: show security flow session session-identifier Output Fields *(continued)*

Field Name	Field Description
Out	For the reverse flow: • Source and destination IP addresses, and application protocol for the reverse flow. • **Interface**: Reverse flow interface. • **Session token**: Internal token derived from the virtual routing instance. • **Flag**: Internal debugging flags. • **Route**: Internal next hop of the route to be used by the flow. • **Gateway**: Next-hop gateway of the flow. • **Tunnel**: Tunnel type identifier. Used for internal debugging. If the flow is going into a tunnel, species the decimal format of the tunnel ID, plus the tunnel type. Otherwise, 0 (zero). See Table 69 on page 686 for tunnel type identification. • **Port Sequence, FIN sequence, FIN state, Cookie**: Internal TCP state tracking information. • **Pkts, Bytes, CP Session ID**: Packets and bytes matched on the wing, and the associated CP session ID of the wing. • **Conn tag**: Session connection tag for GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.
Status	Session status: • Auth (NAT flag with NAT_AUTH) • Transparent (NAT flag with NAT_TRANSPARENT) • Expired (NAT flag with NAT_INVALID) • Normal (no flag)
Virtual system	Virtual system to which the session belongs (it is optional).
Application	Application match for applying the rule. Application: junos-http/6.
Rule	Name of the application traffic control rule. Rule: INVALID.
Current timeout	Remaining time for the session unless traffic exists in the session.
Duration	Length of time, in seconds, for which the session is active.

Table 68: show security flow session session-identifier Output Fields *(continued)*

Field Name	Field Description
Client	Name of the ALG, if there is resource manager.
Group	Group identification number, if there is resource manager.
Resource	Resource identification number, if there is resource manager.

Table 69: Tunnel Type Identification

Binary (first 3 bits)	Hexadecimal (the first 4 bits)	Tunnel Type
0x000	0x0	no tunnel
0x001	0x2	TUNNEL_TYPE_IPSEC
0x010	0x4	TUNNEL_TYPE_L2TP
0x011	0x6	TUNNEL_TYPE_NATT
0x100	0x8	TUNNEL_TYPE_DS_LITE
0x101	0xa	TUNNEL_TYPE_MCNH

Sample Output

show security flow session session-identifier 20595

`root>` **show security flow session session-identifier 20595**

```
Flow Sessions on FPC10 PIC2:

Session ID: 20595, Status: Normal
Flags: 0x42/0x0/0x1/0x8103
Policy name: pre-id-default-policy-logical-system-00/3
Source NAT pool: Null, Application: junos-http/6
Dynamic application: INCONCLUSIVE,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
```

```
Maximum timeout: 1800, Current timeout: 1788
Session State: Valid
Start time: 247736, Duration: 14
   In: 4.0.0.1/47931 --> 5.0.0.2/80;tcp,
    Conn Tag: 0x0, Interface: ge-0/0/2.0,
    Session token: 0x7, Flag: 0x3621
    Route: 0xe0010, Gateway: 4.0.0.1, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 3, Bytes: 176
    Out: 5.0.0.2/80 --> 4.0.0.1/47931;tcp,
    Conn Tag: 0x0, Interface: ge-0/0/3.0,
     Session token: 0x8, Flag: 0x3620
    Route: 0xc0010, Gateway: 5.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 2, Bytes: 120
 Total sessions: 1
```

show security flow session source-port

Syntax

```
show security flow session source-port
source-port-number
[brief | extensive | summary]
```

Release Information

Command introduced in Junos OS Release 8.5; Filter and view options introduced in Junos OS Release 10.2.

Description

Display information about each session that uses the specified source port.

Options

- ***source-port-number*** —Number of the source port about which to display sessions information.
- brief | extensive | summary–Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow session source-port | 443

List of Sample Output

show security flow session source-port 15198 on page 690
show security flow session source-port 15198 brief on page 690
show security flow session source-port 15198 extensive on page 691
show security flow session source-port 15198 summary on page 692

Output Fields

Table 70 on page 689 lists the output fields for the **show security flow session source-port** command. Output fields are listed in the approximate order in which they appear.

Table 70: show security flow session source-port Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
Resource information	Information about the session particular to the resource manager, including the name of the ALG, the group ID, and the resource ID.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Valid sessions	Number of valid sessions.
Pending sessions	Number of pending sessions.

Table 70: show security flow session source-port Output Fields *(continued)*

Field Name	Field Description
Invalidated sessions	Number of invalidated sessions.
Sessions in other states	Number of sessions in other states.

Sample Output

show security flow session source-port 15198

```
root> show security flow session source-port 15198
```

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Policy name: p1/4, Timeout: 770, Valid
  In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp, If: ge-7/1/0.0, Pkts: 9, Bytes: 507,
CP Session ID: 420000002
  Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp, If: ge-7/1/1.0, Pkts: 8, Bytes: 462,
 CP Session ID: 420000002
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session source-port 15198 brief

```
root> show security flow session source-port 15198 brief
```

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Policy name: p1/4, Timeout: 740, Valid
  In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp, If: ge-7/1/0.0, Pkts: 9, Bytes: 507,
CP Session ID: 420000002
  Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp, If: ge-7/1/1.0, Pkts: 8, Bytes: 462,
```

```
 CP Session ID: 420000002
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session source-port 15198 extensive

root> **show security flow session source-port 15198 extensive**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Status: Normal
Flags: 0x40/0x0/0x2408003, services-offload
Policy name: p1/4
Source NAT pool: Null, Application: junos-telnet/10
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 750
Session State: Valid
Start time: 165, Duration: 1056
   In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0x42001021
    Route: 0x80010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 507
    CP Session ID: 420000002
   Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x42001020
    Route: 0x70010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 462
    CP Session ID: 420000002
Total sessions: 1
```

```
Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session source-port 15198 summary

`root>` **show security flow session source-port 15198 summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow session source-prefix

Syntax

```
show security flow session source-prefix
source-prefix-number
[brief | extensive | summary]
```

Release Information

Command introduced in Junos OS Release 8.5.
Support for IPv6 addresses added in Junos OS Release 10.2. Support for IPv6 addresses in active/active chassis cluster configurations (in addition to the existing support of active/passive chassis cluster configurations) added in Junos OS Release 10.4.
Filter and view options introduced in Junos OS Release 10.2.

Description

Display information about each session that uses the specified source prefix.

Options

source-prefix-number—Source IP prefix or address for which to display sessions information.

brief | extensive | summary—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

clear security flow session source-prefix | 445

List of Sample Output

show security flow session source-prefix 200.0.0.10 on page 695
show security flow session source-prefix 200.0.0.10 brief on page 695
show security flow session source-prefix 200.0.0.10 extensive on page 696
show security flow session source-prefix 200.0.0.10 summary on page 697

Output Fields

Table 71 on page 694 lists the output fields for the **show security flow session source-prefix** command. Output fields are listed in the approximate order in which they appear.

Table 71: show security flow session source-prefix Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic.
Timeout	Idle timeout after which the session expires.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Out	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flag	Internal flag depicting the state of the session, used for debugging purposes.
Policy name	Name and ID of the policy that the first packet of the session matched.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Valid sessions	Number of valid sessions.
Pending sessions	Number of pending sessions.
Invalidated sessions	Number of invalidated sessions.

Table 71: show security flow session source-prefix Output Fields *(continued)*

Field Name	Field Description
Sessions in other states	Number of sessions in other states.

Sample Output

show security flow session source-prefix 200.0.0.10

root> **show security flow session source-prefix 200.0.0.10**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Policy name: p1/4, Timeout: 488, Valid
  In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp, If: ge-7/1/0.0, Pkts: 9, Bytes: 507,
CP Session ID: 420000002
  Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp, If: ge-7/1/1.0, Pkts: 8, Bytes: 462,
 CP Session ID: 420000002
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session source-prefix 200.0.0.10 brief

root> **show security flow session source-prefix 200.0.0.10 brief**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Policy name: p1/4, Timeout: 482, Valid
  In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp, If: ge-7/1/0.0, Pkts: 9, Bytes: 507,
CP Session ID: 420000002
  Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp, If: ge-7/1/1.0, Pkts: 8, Bytes: 462,
 CP Session ID: 420000002
Total sessions: 1
```

```
Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session source-prefix 200.0.0.10 extensive

root> **show security flow session source-prefix 200.0.0.10 extensive**

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Session ID: 420000002, Status: Normal
Flags: 0x40/0x0/0x2408003, services-offload
Policy name: p1/4
Source NAT pool: Null, Application: junos-telnet/10
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: 1800, Current timeout: 436
Session State: Valid
Start time: 165, Duration: 1370
   In: 200.0.0.10/15198 --> 60.0.0.2/23;tcp,
    Interface: ge-7/1/0.0,
    Session token: 0x6, Flag: 0x42001021
    Route: 0x80010, Gateway: 200.0.0.10, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 9, Bytes: 507
    CP Session ID: 420000002
   Out: 60.0.0.2/23 --> 200.0.0.10/15198;tcp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x42001020
    Route: 0x70010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 8, Bytes: 462
    CP Session ID: 420000002
Total sessions: 1

Flow Sessions on FPC10 PIC3:
Total sessions: 0
```

show security flow session source-prefix 200.0.0.10 summary

root> **show security flow session source-prefix 200.0.0.10 summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow session summary family

Syntax

```
show security flow session summary family (inet | inet6)
```

Release Information

Command introduced in Junos OS Release 10.2.
Support on SRX Series devices for flow-based mode for family inet6 added in Junos OS Release 10.2.
Support for IPv6 addresses in active/active chassis cluster configurations (in addition to the existing support of active/passive chassis cluster configurations) added in Junos OS Release 10.4.

Description

Display filtered summary of information about existing sessions, including types of sessions, active and failed sessions, and the maximum allowed number of sessions.

Options

- **inet**—Display details summary of IPv4 sessions.
- **inet6**—Display details summary of IPv6 sessions.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
clear security flow session all | 405

List of Sample Output

show security flow session summary family inet on page 699
show security flow session summary family inet6 on page 700

Output Fields

Table 72 on page 698 lists the output fields for the **show security flow session summary family** command. Output fields are listed in the approximate order in which they appear.

Table 72: show security flow session summary Output Fields

Field Name	Field Description
Valid sessions	Count of valid sessions.

Table 72: show security flow session summary Output Fields *(continued)*

Field Name	Field Description
Pending sessions	Count of pending sessions.
Invalidated sessions	Count of sessions the security device has determined to be invalid.
Sessions in other states	Count of sessions not in valid, pending, or invalidated state.
Total sessions	Total of the above counts.

Sample Output

show security flow session summary family inet

`user@host>` **show security flow session summary family inet**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1

Flow Sessions on FPC10 PIC3:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0
```

show security flow session summary family inet6

```
user@host> show security flow session summary family inet6
```

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
```

show security flow session summary node

Syntax

```
show security flow session summary node (node-id | all | local | primary)
```

Release Information

Command introduced in Junos OS Release 8.5; node options added in Junos OS Release 9.0. Filter options added in Junos OS Release 10.2.

Support on SRX Series devices for flow-based mode for family inet6 added in Junos OS Release 10.2.

Support for IPv6 addresses in active/active chassis cluster configurations (in addition to the existing support of active/passive chassis cluster configurations) added in Junos OS Release 10.4.

Description

Display information about all currently active security sessions on the device for the specified node options in summary mode.

Options

node—(Optional) For chassis cluster configurations, display session information on a specific node.

- ***node-id*** —Identification number of the node. It can be 0 or 1.
- **all**—Display information about all nodes.
- **local**—Display information about the local node.
- **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

show security flow session | 610

List of Sample Output

show security flow session summary node 0 on page 702
show security flow session summary node 1 on page 703
show security flow session summary node all on page 704
show security flow session summary node local on page 706
show security flow session summary node primary on page 707

Output Fields

Table 73 on page 702 lists the output fields for the **show security flow session summary node** command. Output fields are listed in the approximate order in which they appear.

Table 73: show security flow session summary node Output Fields

Field Name	Field Description
Unicast-sessions	Number of unicast sessions.
Multicast-sessions	Number of multicast sessions.
Failed-sessions	Number of failed sessions.
Sessions-in-use	Number of sessions in use. • Valid sessions • Pending sessions • Invalidated sessions • Sessions in other states
Maximum-sessions	Maximum number of sessions permitted.

Sample Output

show security flow session summary node 0

```
root@host> show security flow session summary node 0
```

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

```
Flow Sessions on FPC0 PIC2:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC3:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

show security flow session summary node 1

`root@host>` **show security flow session summary node 1**

```
node1:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

```
Flow Sessions on FPC0 PIC2:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC3:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

show security flow session summary node all

`root@host>` **show security flow session summary node all**

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC2:
```

```
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC3:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

node1:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC2:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
```

```
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC3:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

show security flow session summary node local

root@host> **show security flow session summary node local**

```
node0:
--------------------------------------------------------------------------


Flow Sessions on FPC0 PIC1:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC2:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
```

```
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC3:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

show security flow session summary node primary

root@host> **show security flow session summary node primary**

```
node0:
--------------------------------------------------------------------------

Flow Sessions on FPC0 PIC1:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC2:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
```

```
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456

Flow Sessions on FPC0 PIC3:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 6291456
```

show security flow session summary services-offload

Syntax

```
show security flow session summary services-offload [filter]
```

Release Information

Command introduced in Junos OS Release 11.4.

Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the SRX5K-MPC3-100G10G (IOC3) and the SRX5K-MPC3-40G10G (IOC3) with Express Path (formerly known as *services offloading*) support are introduced for SRX5400, SRX5600, and SRX5800 devices.

This command is supported on the SRX1500, SRX 5800, SRX 5600, and SRX 5400 devices, and vSRX.

Description

Display information about all currently active services-offload security sessions on the device in summary mode.

Options

- *filter*—Filter the display by the specified criteria.

  The following filters reduce the display to those sessions that match the criteria specified by the filter:

  application—Application name.

  application-firewall-rule-set—Application firewall enabled with the specified rule set.

  application-traffic-control-rule-set—Application traffic control enabled with the specified rule set.

  destination-port—Destination port.

  destination-prefix—Destination IP prefix or address.

  dynamic-application—Dynamic application name.

  dynamic-application-group—Dynamic application group name.

  family—Protocol family.

  interface—Name of incoming or outgoing interface.

  logcal-system—Logical system name.

  protocol—IP protocol number.

  root-logical-system—Root logical system name.

  source-port—Source port.

  source-prefix—Source IP prefix or address.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25
clear security flow session services-offload | 437

List of Sample Output
show security flow session summary services-offload on page 711
show security flow session summary services-offload application on page 712
show security flow session summary services-offload destination-port on page 712

Output Fields
Table 74 on page 710 lists the output fields for the **show security flow session summary services-offload** command. Output fields are listed in the approximate order in which they appear.

Table 74: show security flow session summary services-offload Output Fields

Field Name	Field Description
Unicast-sessions	Number of unicast sessions.
Multicast-sessions	Number of multicast sessions.
Services-offload-sessions	Number of services-offload sessions.
Failed-sessions	Number of failed sessions.
Sessions-in-use	Number of sessions in use: • Valid • Pending • Invalidated • Sessions in other states
Maximum-sessions	Maximum number of sessions.

Sample Output

show security flow session summary services-offload

`user@host>` **show security flow session summary services-offload**

```
Flow Sessions on FPC1 PIC0:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 409600

Flow Sessions on FPC2 PIC0:
Unicast-sessions: 1
Multicast-sessions: 0
Services-offload-sessions: 1
Failed-sessions: 0
Sessions-in-use: 1
  Valid sessions: 1
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 819200

Flow Sessions on FPC3 PIC0:
Unicast-sessions: 0
Multicast-sessions: 0
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 819200

Flow Sessions on FPC5 PIC0:
Unicast-sessions: 0
Multicast-sessions: 0
```

```
Services-offload-sessions: 0
Failed-sessions: 0
Sessions-in-use: 0
  Valid sessions: 0
  Pending sessions: 0
  Invalidated sessions: 0
  Sessions in other states: 0
Maximum-sessions: 819200
```

show security flow session summary services-offload application

```
user@host> show security flow session summary services-offload application telnet
```

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC2:

Valid sessions: 0
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Valid sessions: 1
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 1
```

show security flow session summary services-offload destination-port

```
user@host> show security flow session summary services-offload destination-port 23
```

```
Flow Sessions on FPC10 PIC1:
Total sessions: 0

Flow Sessions on FPC10 PIC2:
Total sessions: 0

Flow Sessions on FPC10 PIC3:

Session ID: 430000004, Policy name: p1/4, Timeout: 1500, Valid
  In: 200.0.0.10/15200 --> 60.0.0.2/23;tcp, If: ge-7/1/0.0, Pkts: 13, Bytes: 718,
 CP Session ID: 430000003
  Out: 60.0.0.2/23 --> 200.0.0.10/15200;tcp, If: ge-7/1/1.0, Pkts: 12, Bytes: 677,
 CP Session ID: 430000003
Total sessions: 1
```

show security flow session tunnel

Syntax

show security flow session tunnel
[brief | extensive | summary]

Release Information
Command introduced in Junos OS Release 8.5; Filter and view options introduced in Junos OS Release 10.2. Fragmentation counters options introduced in Junos OS Release 15.1X49-90.

> **NOTE:** Only **show security flow session tunnel extensive** and **show security flow session tunnel summary** provide fragmentation counters output.

Description
Display information about all tunnel sessions.

Options
none—Display the **brief** (default) level of output.

brief | extensive | summary—Display the specified level of output.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output
show security flow session tunnel on page 716
show security flow session tunnel brief on page 717
show security flow session tunnel extensive on page 718
show security flow session tunnel summary extensive (with fragmentation counters output) on page 721
show security flow session tunnel summary on page 722
show security flow session tunnel summary (with fragmentation counters output) on page 723

Output Fields
Table 75 on page 715 lists the output fields for the **show security flow session tunnel** command. Output fields are listed in the approximate order in which they appear.

Table 75: show security flow session tunnel Output Fields

Field Name	Field Description
Session ID	Number that identifies the session. You can use this ID to get additional information about the session.
Policy name	Policy that permitted the traffic. NA (Not Applicable) for a tunnel session.
Source NAT pool	The name of the source pool where NAT is used.
Timeout	Idle timeout after which the session expires. NA (Not Applicable) for a tunnel session.
In	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, encapsulation and authentication header fragments generated, inner IPv4 fragments generated, inner IPv6 fragments generated, port sequence, FIN sequence, FIN state, packets and bytes).
Total sessions	Total number of sessions.
Status	Session status.
Flags	Internal flag depicting the state of the session, used for debugging purposes.
Source NAT pool	The name of the source pool where NAT is used.
Application	Name of the application.
Maximum timeout	Maximum session timeout.
Current timeout	Remaining time for the session unless traffic exists in the session.
Encryption	Encryption traffic name.
Session State	Session state.
Start time	Time when the session was created, offset from the system start time.
Session token	Internal token derived from the virtual routing instance.
Route	Internal next hop of the route to be used by the flow.
Valid sessions	Number of valid sessions.

Table 75: show security flow session tunnel Output Fields *(continued)*

Field Name	Field Description
Pending sessions	Number of pending sessions.
Invalidated sessions	Number of invalidated sessions.
Sessions in other states	Number of sessions in other states.
ESP/AH frag Rx: *number*, Generated: *number*	For IPsec tunnels, the number of Encapsulating Security Payload (ESP) or Authentication Header (AH) fragments that were received and the number that were generated.
Inner IPv4 frag Rx: *number*, Tx: *number*, Generated: *number*	For tunnels with IPv4 fragments, the number of fragments associated with the tunnel that were received, transmitted, and generated.
Inner IPv6 frag Rx: *number*, Tx: *number*, Generated: *number*	For tunnels with IPv6 fragments, the number of fragments associated with the tunnel that were received, transmitted, and generated.

Sample Output

show security flow session tunnel

`root>` **show security flow session tunnel**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000001, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/43405 --> 60.0.0.3/494;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP
Session ID: 420000000

Session ID: 410000002, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000
Total sessions: 2

Flow Sessions on FPC10 PIC2:

Session ID: 420000003, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000
```

```
Session ID: 420000004, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;ah, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000
Total sessions: 2

Flow Sessions on FPC10 PIC3:

Session ID: 430000005, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000

Session ID: 430000006, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;ah, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000
Total sessions: 2
```

show security flow session tunnel brief

root> **show security flow session tunnel brief**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000001, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/43405 --> 60.0.0.3/494;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP
Session ID: 420000000

Session ID: 410000002, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000
Total sessions: 2

Flow Sessions on FPC10 PIC2:

Session ID: 420000003, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000

Session ID: 420000004, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;ah, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000
Total sessions: 2

Flow Sessions on FPC10 PIC3:
```

```
Session ID: 430000005, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;esp, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000

Session ID: 430000006, Policy name: N/A, Timeout: N/A, Valid
  In: 60.0.0.2/0 --> 60.0.0.3/0;ah, If: ge-7/1/1.0, Pkts: 0, Bytes: 0, CP Session
 ID: 420000000
Total sessions: 2
```

show security flow session tunnel extensive

root> **show security flow session tunnel extensive**

```
Flow Sessions on FPC10 PIC1:

Session ID: 410000001, Status: Normal
Flags: 0x10000/0x0/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: N/A, Current timeout: N/A
Session State: Valid
Start time: 3548, Duration: 797
   In: 60.0.0.2/43405 --> 60.0.0.3/494;esp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x80100621
    Route: 0x60010, Gateway: 60.0.0.2, Tunnel: 0
    ESP/AH frag Rx: 0, Generated: 0
   Inner IPv4 frag Rx: 4, Tx: 4, Generated: 4,
    Inner IPv6 frag Rx: 0, Tx: 0, Generated: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 420000000

Session ID: 410000002, Status: Normal
Flags: 0x10000/0x0/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
```

```
Maximum timeout: N/A, Current timeout: N/A
Session State: Valid
Start time: 3548, Duration: 797
   In: 60.0.0.2/0 --> 60.0.0.3/0;esp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x621
    Route: 0x60010, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 420000000
Total sessions: 2

Flow Sessions on FPC10 PIC2:

Session ID: 420000003, Status: Normal
Flags: 0x10000/0x0/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: N/A, Current timeout: N/A
Session State: Valid
Start time: 3513, Duration: 798
   In: 60.0.0.2/0 --> 60.0.0.3/0;esp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x621
    Route: 0x0, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 420000000


Session ID: 420000004, Status: Normal
Flags: 0x10000/0x0/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: N/A, Current timeout: N/A
Session State: Valid
```

```
Start time: 3513, Duration: 798
   In: 60.0.0.2/0 --> 60.0.0.3/0;ah,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x621
    Route: 0x0, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 420000000
Total sessions: 2

Flow Sessions on FPC10 PIC3:

Session ID: 430000005, Status: Normal
Flags: 0x10000/0x0/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: N/A, Current timeout: N/A
Session State: Valid
Start time: 3513, Duration: 799
   In: 60.0.0.2/0 --> 60.0.0.3/0;esp,
    Interface: ge-7/1/1.0,
    Session token: 0x7, Flag: 0x621
    Route: 0x0, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 420000000

Session ID: 430000006, Status: Normal
Flags: 0x10000/0x0/0x1
Policy name: N/A
Source NAT pool: Null
Dynamic application: junos:UNKNOWN,
Encryption:  Unknown
Application traffic control rule-set: INVALID, Rule: INVALID
Maximum timeout: N/A, Current timeout: N/A
Session State: Valid
Start time: 3513, Duration: 799
   In: 60.0.0.2/0 --> 60.0.0.3/0;ah,
    Interface: ge-7/1/1.0,
```

```
    Session token: 0x7, Flag: 0x621
    Route: 0x0, Gateway: 60.0.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 420000000
Total sessions: 2
```

show security flow session tunnel summary extensive (with fragmentation counters output)

`root>` **show security flow session tunnel extensive**

```
node0:
Flow Sessions on FPC2 PIC1:
  Session ID: 90000004, Status: Normal, State: Active
  Flags: 0x10000/0x0/0x1
  Policy name: N/A
  Source NAT pool: Null
  Dynamic application: junos:UNKNOWN,
  Encryption:  Unknown
  Application traffic control rule-set: INVALID, Rule: INVALID
  Maximum timeout: N/A, Current timeout: N/A
  Session State: Valid
  Start time: 6251, Duration: 167168
    In: 2.2.2.2/0 --> 2.2.2.1/10203;esp,
    Conn Tag: 0x0, Interface: reth1.0,
    Session token: 0x7, Flag: 0x80100621
    Route: 0x867f3c1, Gateway: 2.2.2.2, Tunnel: 0
    ESP/AH frag Rx: 0, Generated: 0
    Inner IPv4 frag Rx: 27, Tx: 27, Generated: 18,
    Inner IPv6 frag Rx: 0, Tx: 0, Generated: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0,
    Pkts: 0, Bytes: 0
    CP Session ID: 90000000


    Session ID: 90000005, Status: Normal, State: Active
    Flags: 0x10000/0x0/0x1
    Policy name: N/A
    Source NAT pool: Null
    Dynamic application: junos:UNKNOWN,
    Encryption:  Unknown
    Application traffic control rule-set: INVALID, Rule: INVALID
```

```
    Maximum timeout: N/A,
    Current timeout: N/A
    Session State: Valid
    Start time: 6251, Duration: 167168
       In: 2.2.2.2/0 --> 2.2.2.1/0;esp,
       Conn Tag: 0x0, Interface: reth1.0,
       Session token: 0x7, Flag: 0x100621
       Route: 0x867f3c1, Gateway: 2.2.2.2, Tunnel: 0
       ESP/AH frag Rx: 0, Generated: 0
       Inner IPv4 frag Rx: 0, Tx: 0, Generated: 0,
       Inner IPv6 frag Rx: 0, Tx: 0, Generated: 0
       Port sequence: 0,
       FIN sequence: 0,
       FIN state: 0,
       Pkts: 0, Bytes: 0

       CP Session ID: 90000000
      Total sessions: 2
```

show security flow session tunnel summary

root> **show security flow session tunnel summary**

```
Flow Sessions on FPC10 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2

Flow Sessions on FPC10 PIC2:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2

Flow Sessions on FPC10 PIC3:

Valid sessions: 2
Pending sessions: 0
```

```
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2
```

show security flow session tunnel summary (with fragmentation counters output)

`root>` **show security flow session tunnel summary**

```
node0:
Flow Sessions on FPC2 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2

Tunnel fragment summary:
Tunnels with ESP/AH frag Rx: 0 (0)
Tunnels with ESP/AH frag generated: 0 (0)
Tunnels with IPv4 frag Rx: 1 (27)
Tunnels with IPv4 frag Tx: 1 (27)
Tunnels with IPv4 frag generated: 1 (18)
Tunnels with IPv6 frag Rx: 0 (0)
Tunnels with IPv6 frag Tx: 0 (0)
Tunnels with IPv6 frag generated: 0 (0)

Flow Sessions on FPC2 PIC1:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2


Tunnel fragment summary:
Tunnels with ESP/AH frag Rx: 0 (0)
Tunnels with ESP/AH frag generated: 0 (0)
Tunnels with IPv4 frag Rx: 0 (0)
Tunnels with IPv4 frag Tx: 0 (0)
Tunnels with IPv4 frag generated: 0 (0)
Tunnels with IPv6 frag Rx: 0 (0)
Tunnels with IPv6 frag Tx: 0 (0)
```

```
Tunnels with IPv6 frag generated: 0 (0)


Flow Sessions on FPC2 PIC3:

Valid sessions: 2
Pending sessions: 0
Invalidated sessions: 0
Sessions in other states: 0
Total sessions: 2

Tunnel fragment summary:
Tunnels with ESP/AH frag Rx: 0 (0)
Tunnels with ESP/AH frag generated: 0 (0)
Tunnels with IPv4 frag Rx: 0 (0)
Tunnels with IPv4 frag Tx: 0 (0)
Tunnels with IPv4 frag generated: 0 (0)
Tunnels with IPv6 frag Rx: 0 (0)
Tunnels with IPv6 frag Tx: 0 (0)
Tunnels with IPv6 frag generated: 0 (0)


Tunnel fragment summary:
Tunnels with ESP/AH frag Rx: 0 (0)
Tunnels with ESP/AH frag generated: 0 (0)
Tunnels with IPv4 frag Rx: 1 (27)
Tunnels with IPv4 frag Tx: 1 (27)
Tunnels with IPv4 frag generated: 1 (18)
Tunnels with IPv6 frag Rx: 0 (0)
Tunnels with IPv6 frag Tx: 0 (0)
Tunnels with IPv6 frag generated: 0 (0)
```

show security flow statistics

Syntax

```
show security flow statistics
<node (node-id | all | local | primary)>
<logical-system (logical-system-name | all)>
<tenant (tenant-name | all)>
```

Release Information

Command introduced in Junos OS Release 10.2.
Fragmentation counters options introduced in Junos OS Release 15.1X49-90.
Support added at the logical system and tenant system level in Junos OS Release 20.1R1.

Description

Display security flow statistics on a specific SPU. A flow is a stream of related packets that meet the same matching criteria and share the same characteristics.

A packet undergoes flow-based processing after packet-based filters and some screens have been applied to it. A System Processing Unit (SPU) processes the packets of a flow according to the security features and other services configured for the session. Flow-based packet processing treats related packets, or a stream of packets, in the same way. Packet treatment depends on characteristics that were established for the first packet of the packet stream.

The **show security flow statistics** command displays information for individual SPUs. For each SPU, the active sessions on the SPU, packets received, packets transmitted, packets forwarded/queued, packets copied, packets dropped, packet fragments received in a flow on the SPU, pre-fragmented packets generated, and post-fragmented packets generated are displayed in terms of numbers.

There are many conditions that can cause a packet to be dropped. Here are some of them:

- A screen module detects IP spoofing
- The IPSec Encapsulating Security Payload (ESP) or the Authentication Header (AH) authentication failed. For example, incoming NAT errors could cause this to happen.
- A packet matches more than one security policy that specifies user authentication. (Sometimes packets are looped through the system more than once. Each time a packet passes through the system, that packet must be permitted by a policy.)
- A time constraint setting expires. For example, multicast streams with a packet interval of more than 60 seconds would experience premature aging-out of flow sessions. (In most cases, you can configure higher time-out value to prevent packet drop.)

Packet fragmentation can occur for a number of reasons, and, in some cases, it can be controlled through a configuration setting. Every link has a maximum transmission unit (MTU) size that specifies the size of

the largest packet that the link can transmit. A larger MTU size means that fewer packets are required to transmit a certain amount of data. However, for a packet to successfully traverse the path from the source node to the destination node, the MTU size of the source node egress interface must be no larger than that of the smallest MTU size of all nodes on the path between the source and destination. This value is referred to as the path maximum transmission unit (path MTU).

When a packet is larger that the MTU size on any link in the data path, the link might fragment it or drop it.

- For IPv4, if a node within the path between a source node and a destination node receives a packet that is larger than its MTU size, it can fragment the packet and transmit the resulting smaller packets.
- For IPv6, an intermediate node cannot fragment a packet. If a packet is larger than a link's MTU size, it is likely that the link will drop it. However, the source node (the node that sent the packet) can fragment a packet, and this is done to accommodate a path MTU size-adjustment requirement. Nodes along the path of a packet cannot fragment the packet to transmit it.

The fragmentation counters feature for IPsec tunnels provides the show output information for the pre-fragments generated and post-fragments generated fields.

Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) are introduced for SRX5400, SRX5600, and SRX5800 devices that perform hash-based datapath packet forwarding to interconnect with all existing IOC and SPC cards using the XL chip (packet-processing chip). The IOC3 XL chip uses a hash-based method to distribute ingress traffic to a pool of SPUs by default.

Options

none—Displays the security flow statistics information.

node—(Optional) For chassis cluster configurations, displays all security flow statistics on a specific node (device) in the cluster.

- ***node-id***—Identification number of the node. It can be 0 or 1.
- **all**—Displays information about all nodes.
- **local**—Displays information about the local node.
- **primary**—Displays information about the primary node.

logical-system *logical-system-name*—(Optional) Displays information about the specified logical system.

logical-system *all*—(Optional) Displays information about all the logical systems.

tenant *tenant-name*—(Optional) Displays information about the specified tenant system.

tenant *all*—(Optional) Displays information about all tenant systems.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security flow statistics on page 728
show security flow statistics logical-system LSYS1 on page 729
show security flow statistics on page 729
show security flow statistics tenant TSYS1 on page 729
show security flow statistics on page 730

Output Fields

Table 76 on page 727 lists the output fields for the **show security flow statistics** command. Output fields are listed in the approximate order in which they appear.

Table 76: show security flow statistics Output Fields

Field Name	Field Description
Current sessions	Number of active sessions on the SPU.
Packets received	Number of packets received in a security flow of a specific SPU. The packets are processed and forwarded on that SPU.
Packets transmitted	Number of packets returned to Jexec for transmission.
Packets forwarded/queued	Number of packets fowarded or number of packets queued up by other modules. **NOTE:** Dropped packets are not captured by this field.
Packets copied	Number of packets copied by other modules including fragmentation and tcp proxy.
Packets dropped	Number of packets dropped in a flow on a specific SPU. The packets are received in the flow. However, during processing, the system discovers sanity check errors, security violations, or other conditions that caused the packet to be dropped. See the description for some of the conditions and events that can cause a packet to be dropped.

Table 76: show security flow statistics Output Fields *(continued)*

Field Name	Field Description
Fragment packets	Number of fragments received in a flow on the SPU. See the description for information about packet fragments.
Pre fragments generated	For IPsec tunnels, the number of fragments that are self-generated by the SRX Series device before it encapsulates the packet with the IPsec encryption header.
Post fragments generated	For IPsec tunnels, the number of fragments that are received by the SRX Series device and packets that are fragmented after encryption.

Sample Output

show security flow statistics

`user@host>` **show security flow statistics**

```
node0:
--------------------------------------------------------------------------

    Current sessions: 0
    Packets received: 2677
    Packets transmitted: 2278
    Packets forwarded/queued: 0
    Packets copied: 99
    Packets dropped: 300
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0

node1:
--------------------------------------------------------------------------

    Current sessions: 0
    Packets received: 1267
    Packets transmitted: 904
    Packets forwarded/queued: 0
    Packets copied: 0
    Packets dropped: 363
```

```
    Fragment packets: 0
    Pre fragments generated: 0
    Post fragments generated: 0
```

show security flow statistics logical-system LSYS1

```
user@host> show security flow statistics logical-system LSYS1
```

```
Current sessions: 1000
          Packets received: 177888012
          Packets transmitted: 66705150
          Packets forwarded/queued: 14
          Packets copied: 77831798
          Packets dropped: 33351074
          Services-offload packets processed: 16574037502
          Fragment packets: 0
          Pre fragments generated: 0
          Post fragments generated: 0
```

show security flow statistics

```
user@host:LSYS1> show security flow statistics
```

```
Current sessions: 1000
          Packets received: 177888012
          Packets transmitted: 66705150
          Packets forwarded/queued: 14
          Packets copied: 77831798
          Packets dropped: 33351074
          Services-offload packets processed: 16574037502
          Fragment packets: 0
          Pre fragments generated: 0
          Post fragments generated: 0
```

show security flow statistics tenant TSYS1

```
user@host> show security flow statistics tenant TSYS1
```

```
Current sessions: 10
          Packets received: 28711
          Packets transmitted: 10
          Packets forwarded/queued: 0
```

```
            Packets copied: 9590
            Packets dropped: 19541
            Services-offload packets processed: 95909
            Fragment packets: 0
            Pre fragments generated: 0
            Post fragments generated: 0
```

show security flow statistics

user@host:TSYS1> **show security flow statistics**

```
Current sessions: 10
            Packets received: 28711
            Packets transmitted: 10
            Packets forwarded/queued: 0
            Packets copied: 9590
            Packets dropped: 19541
            Services-offload packets processed: 95909
            Fragment packets: 0
            Pre fragments generated: 0
            Post fragments generated: 0
```

show security flow status

Syntax

```
show security flow status
```

Release Information

Command introduced in Junos OS Release 10.2; session distribution mode option added in Junos OS Release 12.1X44-D10; enhanced route scaling mode option added in Junos OS Release 12.1X45-D10. GTP-U distribution option added in Junos OS Release 15.1X49-D40.

Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) are introduced for SRX5400, SRX5600, and SRX5800 devices that perform hash-based data path packet forwarding to interconnect with all existing IOC and SPC cards using the XL chip (packet-processing chip).

The IOC3 XL chip uses a hash-based method to distribute ingress traffic to a pool of SPUs by default. Selection of hash keys depends on application protocols.

Starting in Junos OS Release 18.3R1, flow PowerMode IPsec support is introduced on vSRX instances.

Starting in Junos OS Release 18.4R1, flow PowerMode IPsec support is introduced on SRX4100 and SRX 4200 devices.

Starting in Junos OS Release 18.2R2, flow PowerMode IPsec support is introduced on SRX5400, SRX5600, and SRX5800 devices.

Starting in Junos OS Release 19.1R1, flow PowerMode IPsec support is introduced on SRX4600 devices.

Starting in Junos OS Release 19.4R1, symmetric fat tunnel support is introduced on SRX5400, SRX5600, SRX5800, and vSRX instances.

Description

Display the flow processing modes and logging status.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security flow status on page 733
show security flow status (IPsec Performance Acceleration) on page 733
show security flow status (for hash-based datapath forwarding using SRX5K-MPC3-40G10G (IOC3) and SRX5K-MPC3-100G10G (IOC3) on page 733
show security flow status (SRX5400, SRX5600, SRX5600, and vSRX) on page 734

show security flow status (Tap mode enabled with IP-IP or GRE tunnel inspection on SRX300, SRX320, SRX340, SRX345, SRX1500, SRX4100, and SRX4200) on page 735

Output Fields

Table 77 on page 732 lists the output fields for the **show security flow status** command. Output fields are listed in the approximate order in which they appear.

Table 77: show security flow status Output Fields

Field Name	Field Description
Flow forwarding mode	Flow processing mode. • Inet forwarding mode • Inet6 forwarding mode • MPLS forwarding mode • ISO forwarding mode • Session distribution mode • Enhanced route scaling mode • Tap mode: enabled, inspect ipip
Flow trace status	Flow logging status. • Flow tracing status • Flow tracing options
flow session distribution	SPU load distribution mode. • RR-based • Hash-based GTP-U distribution • Enabled
Flow packet ordering	packet-ordering mode. • Hardware • Software
Flow ipsec performance acceleration	IPsec VPN performance acceleration status.
Flow power mode IPsec	Flow PowerMode IPsec status.

Sample Output

show security flow status

user@host> **show security flow status**

```
      Flow forwarding mode:
        Inet forwarding mode: flow based
        Inet6 forwarding mode: flow based
        MPLS forwarding mode: drop
        ISO forwarding mode: drop
        Tap mode: disabled (default)
      Flow trace status
        Flow tracing status: off
      Flow session distribution
        Distribution mode: Hash-based
        GTP-U distribution: Disabled
      Flow ipsec performance acceleration: off
      Flow packet ordering
        Ordering mode: Hardware
      Flow power mode IPsec: Enabled
```

show security flow status (IPsec Performance Acceleration)

user@host> **show security flow status**

```
Flow forwarding mode:
    Inet forwarding mode: flow based
    Inet6 forwarding mode: drop
    MPLS forwarding mode: drop
    ISO forwarding mode: drop
    TAP mode: enabled
   Flow trace status
   Flow tracing status: off
   Flow session distribution
    Distribution mode: RR-based
    GTP-U distribution: Enabled Flow packet ordering
    Ordering mode: Software (reboot needed to change to software)
   Flow power mode IPsec: Enabled
```

show security flow status (for hash-based datapath forwarding using SRX5K-MPC3-40G10G (IOC3) and SRX5K-MPC3-100G10G (IOC3)

user@host> **show security flow status**

```
node0:
--------------------------------------------------------------------------
   Flow forwarding mode:
    Inet forwarding mode: flow based
    Inet6 forwarding mode: flow based
    MPLS forwarding mode: drop
    ISO forwarding mode: drop
    Tap mode: disabled (default)
  Flow trace status
    Flow tracing status: off
  Flow session distribution
    Distribution mode: Hash-based
    GTP-U distribution: Disabled
  Flow ipsec performance acceleration: off
  Flow packet ordering
    Ordering mode: Hardware
  Flow power mode IPsec: Enabled
  Fat core group status: on
node1:
--------------------------------------------------------------------------
    Flow forwarding mode:
    Inet forwarding mode: flow based
    Inet6 forwarding mode: flow based
    MPLS forwarding mode: drop
    ISO forwarding mode: drop
    Tap mode: disabled (default)
  Flow trace status
    Flow tracing status: off
  Flow session distribution
    Distribution mode: Hash-based
    GTP-U distribution: Disabled
  Flow ipsec performance acceleration: off
  Flow packet ordering
    Ordering mode: Hardware
  Flow power mode IPsec: Enabled
  Fat core group status: on
```

show security flow status (SRX5400, SRX5600, SRX5600, and vSRX)

`user@host>` **show security flow status**

```
  Flow forwarding mode:
    Inet forwarding mode: flow based
    Inet6 forwarding mode: flow based
```

```
    MPLS forwarding mode: drop
    ISO forwarding mode: drop
    Tap mode: disabled (default)
  Flow trace status
    Flow tracing status: off
  Flow session distribution
    Distribution mode: Hash-based
    GTP-U distribution: Disabled
  Flow ipsec performance acceleration: off
  Flow packet ordering
    Ordering mode: Hardware
  Flow power mode IPsec: Disabled
  Fat core group status: on
```

show security flow status (Tap mode enabled with IP-IP or GRE tunnel inspection on SRX300, SRX320, SRX340, SRX345, SRX1500, SRX4100, and SRX4200)

`user@host>` **show security flow status**

```
 Flow forwarding mode:
Inet forwarding mode: flow based
Inet6 forwarding mode: flow based
MPLS forwarding mode: drop
ISO forwarding mode: drop
Tap mode: enabled, inspect ipip
…
```

show security forwarding-options mirror-filter

Syntax

```
show security forwarding-options mirror-filter (all | filter-name)
```

Release Information

Command introduced in Junos OS Release 12.1X46-D10.

Description

Displays status information about all configured mirror filters or that of a specific mirror filter. Each mirror filter contains a set of parameters against which traffic is matched. For each mirror filter, the output identifies the number of packets that were matched by the filter for mirroring and the number of packets that were sent to the packet analyzer. It also shows the parameters that were configured for the mirror filter.

Network operators need a way to monitor X2 traffic to debug any handover issues across eNodeBs. The mirror filter feature allows you to do that. To use the mirror filter feature to monitor X2 traffic, you configure mirror filters. Traffic coming out of an IPsec tunnel is decrypted, mirrored, and analyzed by a packet analyzer, and then encrypted again to go into the outbound IPsec tunnel.

> **NOTE:** The SRX Series mirror filter feature is bidirectional, much like a session. X2 traffic flowing through an IPSec VPN from devices that match the configured filter conditions is mirrored and analyzed.

Starting in Junos OS Release 18.4R1, if the output X2 interface of a mirror filter is configured for an st0 interface to filter traffic that you want to analyze, the packet is duplicated and encrypted by the IPsec tunnel bound to the st0 interface. This enhancement supports the SRX Series devices to send traffic mirrored from a port on an IPsec tunnel.

You can configure up to 15 different mirror filters to be used concurrently.

> **NOTE:** Although there is no minimum required number of parameters for a mirror filter, please be mindful that if you specify too few criteria or accidentally commit an incomplete filter, an over-proportional amount of traffic flow through the system could be mirrored.

Options

all—Display counters for all mirror filters.

filter-name—Name of the mirror filter for which the counters are displayed.

Required Privilege Level
view

RELATED DOCUMENTATION

mirror-filter (Security Forwarding Options) | 346
clear security forward-options mirror filter | 450

List of Sample Output
show security forward-options mirror-filter on page 738

Output Fields
Lists the output fields for the **show security forward-options mirror-filter** command. Output fields are listed in the approximate order in which they appear in the output.

Table 78: show security forward-options mirror-filter

Field Name	Field Description
mirror-filter-name	Name of the mirror filter configured on the device.
interface-in	Name of the incoming logical interface to be matched for mirroring.
interface-out	Name of the outgoing logical interface to be matched for mirroring.
protocol	Networking protocol name or number to be matched for mirroring.
source-port	Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) source port number to be matched for mirroring.
source-prefix	Source IP prefix or address to be matched for mirroring.
destination-port	Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) destination port number to be matched for mirroring.
destination-prefix	Destination IP prefix or address to be matched for mirroring.
filter-counters	Number of packets matched for mirroring.
output-counter	Number of packets sent to the packet analyzer.

Sample Output

show security forward-options mirror-filter

user@host> **show security forward-options mirror-filter traffic-https**

```
Security mirror status

  mirror-filter-name: traffic-https
  interface-in: st0.1
  interface-out: st0.2
  destination-port: 443
  protocol: 132
  source-prefix: 192.0.2.0
  destination-prefix: 203.0.113.0
  filter-counters: 2
  output-counters: 2
```

show security monitoring

Syntax

```
show security monitoring
```

Release Information

Command introduced in Junos OS Release 10.2.

Description

Displays a count of security flow and central point (CP) sessions, CPU utilization (as a percentage of maximum), and memory in use (also as a percentage of maximum) at the moment the command is run. This command is supported on SRX1400, SRX1500, SRX5400, SRX5600, and SRX5800 devices and vSRX.

Required Privilege Level

View

RELATED DOCUMENTATION

- *show security monitoring fpc fpc-number*
- *show security monitoring performance session*
- *show security monitoring performance spu*

show security monitoring

user@host>show security monitoring

```
user@host> show security monitoring
-------------------------------------------------------------------------------
                 Flow session   Flow session     CP session     CP session
FPC PIC CPU Mem        current        maximum        current        maximum
-------------------------------------------------------------------------------
  1   0   0  11              0              0              0              0
  1   1   0   5              3        6291456              1        7549747
  1   2   0   5              2        6291456              0        7549747
  1   3   0   5              3        6291456              1        7549747
  8   0   0  65              4           6963              2           8355
  8   1   0  65              2           6963              0           8355
```

```
Total Sessions:             14       18888294             4       22665951
```

show security monitoring (SRX1400)

user@host>show security monitoring

```
user@host> show security monitoring
-------------------------------------------------------------------------------
                  Flow session   Flow session     CP session     CP session
FPC PIC CPU Mem        current        maximum        current        maximum
-------------------------------------------------------------------------------
  1   0   0  75              0        1048576            0            1048576
```

show security monitoring (vSRX)

user@host>show security monitoring

```
user@host> show security monitoring
-------------------------------------------------------------------------------
                  Flow session   Flow session     CP session     CP session
FPC PIC CPU Mem        current        maximum        current        maximum
-------------------------------------------------------------------------------
  0   0   0  68              2         524288            N/A            N/A
```

show security monitoring (vSRX in a Chassis Cluster)

user@host>show security monitoring

```
node0:
--------------------------------------------------------------------------
                  Flow session   Flow session     CP session     CP session
FPC PIC CPU Mem        current        maximum        current        maximum
--------------------------------------------------------------------------
```

```
  0   0   0  67             0         524288            N/A            N/A

node1:
--------------------------------------------------------------------------
                  Flow session   Flow session     CP session     CP session
FPC PIC CPU Mem         current        maximum        current        maximum
--------------------------------------------------------------------------
  0   0   0  67             0         524288            N/A            N/A
```

show security policies

Syntax

```
show security policies
<all-logical-systems-tenants>
<checksum>
<count>
<detail>
<from-zone zone-name>
<global>
<hit-count>
<information>
<logical-system logical-system-name>
<policy-name policy-name>
<root-logical-system>
<service-set>
<start>
<tenant tenant-name>
<to-zone zone-name>
<unknown-source-identity>
<zone-context>
```

Release Information

Command modified in Junos OS Release 9.2.
Support for IPv6 addresses is added in Junos OS Release 10.2.
Support for wildcard addresses is added in Junos OS Release 11.1.
Support for global policy and services offloading is added in Junos OS Release 11.4.
Support for source-identities and the **Description** output field is added in Junos OS Release 12.1.
Support for negated address added in Junos OS Release 12.1X45-D10.
The output fields for Policy Statistics expanded, and the output fields for the **global** and **policy-name** options are expanded to include from-zone and to-zone global match criteria in Junos OS Release 12.1X47-D10.
Support for the **initial-tcp-mss** and **reverse-tcp-mss** options is added in Junos OS Release 12.3X48-D20.
Output field and description for **source-end-user-profile** option is added in Junos OS Release 15.1x49-D70.
Output field and description for **dynamic-applications** option is added in Junos OS Release 15.1x49-D100.
Output field and description for **dynapp-redir-profile** option is added in Junos OS Release 18.2R1.
The **tenant** option is introduced in Junos OS Release 18.3R1.
The **<all-logical-systems-tenants>** option is introduced in Junos OS Release 18.4R1.
The **information** option is introduced in Junos OS Release 18.4R1.
The **checksum** option is introduced in Junos OS Release 18.4R1.

Description

Displays a summary of all security policies configured on the device. If a particular policy is specified, display information specific to that policy. The existing show commands for displaying the policies configured with multiple tenant support are enhanced. A security policy controls the traffic flow from one zone to another zone. The security policies allow you to deny, permit, reject (deny and send a TCP RST or ICMP port unreachable message to the source host), encrypt and decrypt, authenticate, prioritize, schedule, filter, and monitor the traffic attempting to cross from one security zone to another.

Options

- **all-logical-systems-tenants**—Displays all multitenancy systems.
- **checksum**—Displays the policy information checksum.
- **count**—Displays the number of policies to show. Range is 1 through 65,535.
- **detail**—(Optional) Displays a detailed view of all of the policies configured on the device.
- **from-zone**—Displays the policy information matching the given source zone.
- **global**—(Optional) Displays the policy information about global policies.
- **hit-count**—Displays the policies hit count.
- **information**—Displays the policy information.
- **logical-system**—Displays the logical system name.
- **policy-name**—(Optional) Displays the policy information matching the given policy name.
- **root-logical-system**—Displays root logical system as default.
- **service-set**—Displays the name of the service set.
- **start**—Displays the policies from a given position. Range is 1 through 65,535.
- **tenant**—Displays the name of the tenant system.
- **to-zone**—Displays the policy information matching the given destination zone.
- **unknown-source-identity**—Displays the unknown-source-identity of a policy.
- **zone-context**—Displays the count of policies in each context (from-zone and to-zone).

Required Privilege Level

view

RELATED DOCUMENTATION

Security Policies Overview

Understanding Security Policy Rules

Understanding Security Policy Elements

Unified Policies Configuration Overview

List of Sample Output

show security policies on page 748
show security policies (Dynamic Applications) on page 748
show security policies policy-name detail on page 749
show security policies (Services-Offload) on page 751
show security policies (Device Identity) on page 751
show security policies detail on page 752
show security policies detail (TCP Options) on page 755
show security policies policy-name (Negated Address) on page 755
show security policies policy-name detail (Negated Address) on page 756
show security policies global on page 756
show security policies detail tenant on page 757

Output Fields

Table 79 on page 744 lists the output fields for the **show security policies** command. Output fields are listed in the approximate order in which they appear.

Table 79: show security policies Output Fields

Field Name	Field Description
From zone	Name of the source zone.
To zone	Name of the destination zone.
Policy-name	Name of the applicable policy.
Description	Description of the applicable policy.
State	Status of the policy: • **enabled**: The policy can be used in the policy lookup process, which determines access rights for a packet and the action taken in regard to it. • **disabled**: The policy cannot be used in the policy lookup process, and therefore it is not available for access control.
Index	Internal number associated with the policy.
Sequence number	Number of the policy within a given context. For example, three policies that are applicable in a from-zoneA-to-zoneB context might be ordered with sequence numbers 1, 2, 3. Also, in a from-zoneC-to-zoneD context, four policies might have sequence numbers 1, 2, 3, 4.

Table 79: show security policies Output Fields *(continued)*

Field Name	Field Description
Source addresses	For standard display mode, the names of the source addresses for a policy. Address sets are resolved to their individual names. For detail display mode, the names and corresponding IP addresses of the source addresses for a policy. Address sets are resolved to their individual address name-IP address pairs.
Destination addresses	Name of the destination address (or address set) as it was entered in the destination zone's address book. A packet's destination address must match this value for the policy to apply to it.
source-end-user-profile	Name of the device identity profile (referred to as **end-user-profile** in the CLI) that contains attributes, or characteristics of a device. Specification of the device identity profile in the **source-end-user-profile** field is part of the device identity feature. If a device matches the attributes specified in the profile and other security policy parameters, then the security policy's action is applied to traffic issuing from the device.
Source addresses (excluded)	Name of the source address excluded from the policy.
Destination addresses (excluded)	Name of the destination address excluded from the policy.
Source identities	One or more user roles specified for a policy.
Applications	Name of a preconfigured or custom application whose type the packet matches, as specified at configuration time. • **IP protocol**: The Internet protocol used by the application—for example, TCP, UDP, ICMP. • **ALG**: If an ALG is explicitly associated with the policy, the name of the ALG is displayed. If **application-protocol ignore** is configured, **ignore** is displayed. Otherwise, **0** is displayed. However, even if this command shows **ALG: 0**, ALGs might be triggered for packets destined to well-known ports on which ALGs are listening, unless ALGs are explicitly disabled or when **application-protocol ignore** is not configured for custom applications. • **Inactivity timeout**: Elapsed time without activity after which the application is terminated. • **Source port range**: The low-high source port range for the session application.
Dynamic Applications	Application identification-based Layer 7 dynamic applications.

Table 79: show security policies Output Fields *(continued)*

Field Name	Field Description
Destination Address Translation	Status of the destination address translation traffic: • **drop translated**—Drop the packets with translated destination addresses. • **drop untranslated**—Drop the packets without translated destination addresses.

Application Firewall	An application firewall includes the following:
	• **Rule-set**—Name of the rule set.
	• **Rule**—Name of the rule.
	  ◦ **Dynamic applications**—Name of the applications.
	  ◦ **Dynamic application groups**—Name of the application groups.
	  ◦ **Action**—The action taken with respect to a packet that matches the application firewall rule set. Actions include the following:
	  ▪ **permit**
	  ▪ **deny**
	• **Default rule**—The default rule applied when the identified application is not specified in any rules of the rule set.

Action or Action-type	• The action taken for a packet that matches the policy's tuples. Actions include the following:
	  ◦ **permit**
	  ◦ **firewall-authentication**
	  ◦ **tunnel ipsec-vpn *vpn-name***
	  ◦ **pair-policy *pair-policy-name***
	  ◦ **source-nat pool *pool-name***
	  ◦ **pool-set *pool-set-name***
	  ◦ **interface**
	  ◦ **destination-nat *name***
	  ◦ **deny**
	  ◦ **reject**
	  ◦ **services-offload**

Session log	Session log entry that indicates whether the **at-create** and **at-close** flags were set at configuration time to log session information.
Scheduler name	Name of a preconfigured scheduler whose schedule determines when the policy is active and can be used as a possible match for traffic.

Table 79: show security policies Output Fields *(continued)*

Field Name	Field Description

Policy statistics	• **Input bytes**—The total number of bytes presented for processing by the device.
	  ◦ **Initial direction**—The number of bytes presented for processing by the device from the initial direction.
	  ◦ **Reply direction**—The number of bytes presented for processing by the device from the reply direction.
	• **Output bytes**—The total number of bytes actually processed by the device.
	  ◦ **Initial direction**—The number of bytes from the initial direction actually processed by the device.
	  ◦ **Reply direction**—The number of bytes from the reply direction actually processed by the device.
	• **Input packets**—The total number of packets presented for processing by the device.
	  ◦ **Initial direction**—The number of packets presented for processing by the device from the initial direction.
	  ◦ **Reply direction**—The number of packets presented for processing by the device from the reply direction.
	• **Output packets**—The total number of packets actually processed by the device.
	  ◦ **Initial direction**—The number of packets actually processed by the device from the initial direction.
	  ◦ **Reply direction**—The number of packets actually processed by the device from the reply direction.
	• **Session rate**—The total number of active and deleted sessions.
	• **Active sessions**—The number of sessions currently present because of access control lookups that used this policy.
	• **Session deletions**—The number of sessions deleted since system startup.
	• **Policy lookups**—The number of times the policy was accessed to check for a match.

dynapp-redir-profile	Displays unified policy redirect profile. See *profile(dynamic-application)*.
Per policy TCP Options	Configured syn and sequence checks, and the configured TCP MSS value for the initial direction, the reverse direction or, both.

Sample Output

show security policies

user@host> **show security policies**

```
From zone: trust, To zone: untrust
Policy: p1, State: enabled, Index: 4, Sequence number: 1
  Source addresses:
  sa-1-ipv4: 198.51.100.11/24
  sa-2-ipv6: 2001:db8:a0b:12f0::1/32
  sa-3-ipv6: 2001:db8:a0b:12f0::22/32
  sa-4-wc:   203.0.113.1/255.255.0.255
  Destination addresses:
  da-1-ipv4: 2.2.2.2/24
  da-2-ipv6: 2001:db8:a0b:12f0::8/32
  da-3-ipv6: 2001:db8:a0b:12f0::9/32
  da-4-wc:   192.168.22.11/255.255.0.255
  Source identities: role1, role2, role4
  Applications: any
  Action: permit, application services, log, scheduled
  Application firewall : my_ruleset1
Policy: p2, State: enabled, Index: 5, Sequence number: 2
  Source addresses:
  sa-1-ipv4:  198.51.100.11/24
  sa-2-ipv6:  2001:db8:a0b:12f0::1/32
  sa-3-ipv6:  2001:db8:a0b:12f0::22/32
  Destination addresses:
  da-1-ipv4:  2.2.2.2/24
  da-2-ipv6:  2001:db8:a0b:12f0::1/32
  da-3-ipv6:  2001:db8:a0b:12f0::9/32
  Source identities: role1, role4
  Applications: any
  Action: deny, scheduled
```

show security policies (Dynamic Applications)

user@host>**show security policies**

```
Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
  Source addresses: any
  Destination addresses: any
```

```
    Applications: any
    Dynamic Applications: junos:YAHOO
    Action: deny, log
  Policy: p2, State: enabled, Index: 5, Scope Policy: 0, Sequence number: 2
    Source addresses: any
    Destination addresses: any
    Applications: any
    Dynamic Applications: junos:web, junos:web:social-networking:facebook,
    junos:TFTP, junos:QQ
    Action: permit, log
  Policy: p3, State: enabled, Index: 6, Scope Policy: 0, Sequence number: 3
    Source addresses: any
    Destination addresses: any
    Applications: any
    Dynamic Applications: junos:HTTP, junos:SSL
    Action: permit, application services, log
```

The following example displays the output with unified policies configured.

`user@host>` **show security policies**

```
Default policy: deny-all
Pre ID default policy: permit-all
From zone: trust, To zone: untrust
  Policy: p2, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Applications: junos-defaults
    Dynamic Applications: junos:GMAIL, junos:FACEBOOK-CHAT
    dynapp-redir-profile: profile1
```

show security policies policy-name detail

`user@host>` **show security policies policy-name p1 detail**

```
  Policy: p1, action-type: permit, State: enabled, Index: 4, Scope Policy: 0
  Description: The policy p1 is for the sales team
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses:
    sa-1-ipv4:  198.51.100.11/24
```

```
    sa-2-ipv6:  2001:db8:a0b:12f0::1/32
    sa-3-ipv6:  2001:db8:a0b:12f0::9/32
    sa-4-wc:    203.0.113.1/255.255.0.255
  Destination addresses:
    da-1-ipv4: 192.0.2.0/24
    da-2-ipv6: 2001:db8:a0b:12f0::1/32
    da-3-ipv6: 2001:db8:a0b:12f0::9/32
    da-4-wc:   192.168.22.11/255.255.0.255
Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Destination Address Translation: drop translated
  Application firewall :
  Rule-set: my_ruleset1
    Rule: rule1
      Dynamic Applications: junos:FACEBOOK-ACCESS, junos:YMSG
      Dynamic Application groups: junos:web, junos:chat
      Action: deny
    Default rule: permit
  Session log: at-create, at-close
  Scheduler name: sch20
  Per policy TCP Options: SYN check: No, SEQ check: No
  Policy statistics:
    Input  bytes       :               18144                 545 bps
      Initial direction:                9072                272 bps
      Reply direction  :                9072                 272 bps
    Output bytes       :               18144                 545 bps
      Initial direction:                9072                 272 bps
      Reply direction  :                9072                 272 bps
    Input  packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
    Output packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
    Session rate       :                 108                   3 sps
    Active sessions    :                  93
    Session deletions  :                  15
    Policy lookups     :                 108
```

The following example displays the output with unified policies configured.

user@host> **show security policies policy-name p1 detail**

```
Default policy: permit-all
Pre ID default policy: permit-all
From zone: trust, To zone: trust
  Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Applications: any
    Action: reject
    dynapp-redir-profile: profile1
```

show security policies (Services-Offload)

user@host> **show security policies**

```
  Policy: p1, action-type: reject, State: enabled, Index: 4, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 1
  From zone: trust, To zone: trust
  Source addresses:
    any-ipv4(global): 0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global): 0.0.0.0/0
    any-ipv6(global): ::/0
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  dynapp-redir-profile: profile1(1)
  Per policy TCP Options: SYN check: No, SEQ check: No, Window scale: No
```

show security policies (Device Identity)

user@host> **show security policies**

```
From zone: trust, To zone: untrust
      Policy: dev-id-marketing, State: enabled, Index: 5, Scope Policy: 0, Sequence
```

```
 number: 1
         Source addresses: any
         Destination addresses: any
         source-end-user-profile: marketing-profile
         Applications: any
         Action: permit
```

show security policies detail

user@host> **show security policies detail**

```
Default policy: deny-all
Policy: p1, action-type: permit, services-offload:enabled , State: enabled, Index:
 4, Scope Policy: 0
  Policy Type: Configured
  Description: The policy p1 is for the sales team
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
 Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
  Policy statistics:
    Input  bytes       :               18144                 545 bps
      Initial direction:                9072                272 bps
      Reply direction  :                9072                 272 bps
    Output bytes       :               18144                 545 bps
      Initial direction:                9072                 272 bps
      Reply direction  :                9072                 272 bps
    Input  packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
```

```
    Output packets     :                 216                   6 pps
      Initial direction:                 108                   3 bps
      Reply direction  :                 108                   3 bps
    Session rate       :                 108                   3 sps
    Active sessions    :                  93
    Session deletions  :                  15
    Policy lookups     :                 108
Policy: p2, action-type: permit, services-offload:enabled , State: enabled, Index:
 5, Scope Policy: 0
  Policy Type: Configured
  Description: The policy p2 is for the sales team
  Sequence number: 1
  From zone: untrust, To zone: trust
  Source addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global):  0.0.0.0/0
    any-ipv6(global): ::/0
 Source identities:
  role1
  role2
  role4
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
```

The following example displays the output with unified policies configured.

`user@host>` **show security policies detail**

```
Default policy: deny-all
Pre ID default policy: permit-all
Policy: p2, action-type: reject, State: enabled, Index: 4, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses:
    any-ipv4(global): 0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
```

```
  any-ipv4(global): 0.0.0.0/0
  any-ipv6(global): ::/0
Application: junos-defaults
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [443-443]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [5432-5432]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [80-80]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [3128-3128]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [8000-8000]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [8080-8080]
  IP protocol: 17, ALG: 0, Inactivity timeout: 60
    Source port range: [0-0]
    Destination port range: [1-65535]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [443-443]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [5432-5432]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [80-80]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [3128-3128]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [8000-8000]
  IP protocol: 6, ALG: 0, Inactivity timeout: 1800
    Source port range: [0-0]
    Destination port range: [8080-8080]
  IP protocol: 17, ALG: 0, Inactivity timeout: 60
    Source port range: [0-0]
```

```
      Destination port range: [1-65535]
  Dynamic Application:
    junos:FACEBOOK-CHAT: 10704
    junos:GMAIL: 51
  dynapp-redir-profile: profile1(1)
  Per policy TCP Options: SYN check: No, SEQ check: No, Window scale: No
```

show security policies detail (TCP Options)

user@host> **show security policies policy-name p2 detail**

```
node0:
--------------------------------------------------------------------------
Policy:p2, action-type:permit, State: enabled,Index: 4, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 1
  From zone: trust, To zone: trust
  Source addresses:
    any-ipv4(global): 0.0.0.0/0
    any-ipv6(global): ::/0
  Destination addresses:
    any-ipv4(global): 0.0.0.0/0
    any-ipv6(global): ::/0
  Application: junos-defaults
    IP protocol: tcp, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
 Destination port range: [80-80]
Per policy TCP Options: SYN check: No, SEQ check: No, Window scale: No
  Dynamic-application: junos:HTTP
```

show security policies policy-name (Negated Address)

user@host> **show security policies policy-name p1**

```
node0:
--------------------------------------------------------------------------
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses(excluded): as1
    Destination addresses(excluded): as2
    Applications: any
    Action: permit
```

show security policies policy-name detail (Negated Address)

```
user@host> show security policies policy-name p1 detail
```

```
node0:
--------------------------------------------------------------------------
Policy: p1, action-type: permit, State: enabled, Index: 4, Scope Policy: 0
  Policy Type: Configured
  Sequence number: 1
  From zone: trust, To zone: untrust
  Source addresses(excluded):
    ad1(ad): 255.255.255.255/32
    ad2(ad):  198.51.100.1/24
    ad3(ad):  198.51.100.6  ~ 198.51.100.56
    ad4(ad):  192.0.2.8/24
    ad5(ad):  198.51.100.99 ~ 198.51.100.199
    ad6(ad):  203.0.113.9/24
    ad7(ad):  203.0.113.23/24
  Destination addresses(excluded):
    ad13(ad2): 198.51.100.76/24
    ad12(ad2): 198.51.100.88/24
    ad11(ad2): 192.0.2.23 ~ 192.0.2.66
    ad10(ad2): 192.0.2.93
    ad9(ad2):  203.0.113.76 ~ 203.0.113.106
    ad8(ad2):  203.0.113.199
  Application: any
    IP protocol: 0, ALG: 0, Inactivity timeout: 0
      Source port range: [0-0]
      Destination port range: [0-0]
  Per policy TCP Options: SYN check: No, SEQ check: No
```

show security policies global

```
user@host> show security policies global policy-name Pa
```

```
node0:
--------------------------------------------------------------------------
  Global policies:
  Policy: Pa, State: enabled, Index: 6, Scope Policy: 0, Sequence number: 1
    From zones: any
    To zones: any
    Source addresses: H0
    Destination addresses: H1
    Applications: junos-http
```

```
    Action: permit
```

show security policies detail tenant

`user@host>` **show security policies detail tenant TN1**

```
Default policy: deny-all
Pre ID default policy: permit-all
Policy: p1, action-type: permit, State: enabled, Index: 4, Scope Policy: 0
Policy Type: Configured
Sequence number: 1
From zone: trust, To zone: untrust
Source addresses: any
Destination addresses: any
Application: junos-ping
IP protocol: 1, ALG: 0, Inactivity timeout: 60
ICMP Information: type=255, code=0
Application: junos-telnet
IP protocol: tcp, ALG: 0, Inactivity timeout: 1800
Source port range: [0-0]
Destination port range: [23-23]
Application: app_udp
IP protocol: udp, ALG: 0, Inactivity timeout: 1800
Source port range: [0-0]
Destination port range: [5000-5000]
Application: junos-icmp6-all
IP protocol: 58, ALG: 0, Inactivity timeout: 60
ICMP Information: type=255, code=0
Per policy TCP Options: SYN check: No, SEQ check: No, Window scale: No
Session log: at-create, at-close
Policy statistics:
Input  bytes     :                   0                   0 bps
Initial direction:                   0                   0 bps
Reply direction  :                   0                   0 bps
Output bytes     :                   0                   0 bps
Initial direction:                   0                   0 bps
Reply direction  :                   0                   0 bps
Input  packets   :                   0                   0 pps
Initial direction:                   0                   0 bps
Reply direction  :                   0                   0 bps
Output packets   :                   0                   0 pps
Initial direction:                   0                   0 bps
```

```
Reply direction  :                  0                   0 bps
Session rate     :                  0                   0 sps
Active sessions  :                  0
Session deletions:                  0
Policy lookups   :                  0
```

show security policies hit-count

Syntax

```
show security policies hit-count
<ascending>
<descending>
<from-zone zone-name>
<greater-than count>
<less-than count>
<logical-system (logical-system-name | all)>
<root-logical-system>
<tenant (tenant-name | all)>
<to-zone zone-name>
```

Release Information

Command introduced in Junos OS Release 12.1.
The **index** output field is added to the **show security policies hit-count** command to display the number of sessions redirected in Junos OS Release 18.2R1.
The **tenant** option is introduced in Junos OS Release 18.3R1.

Description

Display the utility rate of security policies according to the number of hits they receive. The number of hits can be listed without an order or sorted in either ascending or descending order, and they can be restricted to the number of hits that fall above or below a specific count or within a range. Data is shown for all zones associated with the policies or named zones.

In a cluster, the count is a sum of all the Services Processing Cards (SPC) hit counts; it is cluster-wide. If a Packet Forwarding Engine (PFE) in a node is in failover mode, but does not reboot, the counter persists. If a node reboots, the PFE in the node also reboots, and the counter is cleared. During an in-service software upgrade (ISSU), all PFEs reboot, therefore all counters are cleared.

Use this command without options to display the number of hits in random order for all security policies and for all zones.

Options

- **ascending**—(Optional) Displays the number of hits for security policies in ascending order.
- **descending**—(Optional) Displays the number of hits for security policies in descending order.
- **from-zone *zone-name***—(Optional) Displays the number of hits for security policies associated with the named source zone.
- **greater-than *count***—(Optional) Displays security policies for which the number of hits is greater than the specified number.

Range: 0 through 4,294,967,295

- **less-than *count***—(Optional) Displays security policies for which the number of hits is less than the specified number.

  Range: 0 through 4,294,967,295

- **logical-system**—Displays the logical system name.
- **root-logical-system**—Displays root logical system as default.
- **tenant**—Displays the name of the tenant system.
- **to-zone *zone-name***—(Optional) Displays the number of hits for security policies associated with the named destination zone.

Required Privilege Level
view

RELATED DOCUMENTATION

> *clear security policies hit-count*
>
> *Security Policies Overview*
>
> *Unified Policies Configuration Overview*

List of Sample Output

show security policies hit-count on page 761
show security policies hit-count ascending on page 761
show security policies hit-count descending greater-than 70 less-than 100 on page 762
show security policies hit-count from-zone untrust to-zone trust on page 762
show security policies hit-count tenant all on page 762

Output Fields

Table 80 on page 760 lists the output fields for the **show security policies hit-count** command. Output fields are listed in the approximate order in which they appear.

Table 80: show security policies hit-count Output Fields

Field Name	Field Description
index	Displays the policy sequence number
from-zone	Name of the source zone
to-zone	Name of the destination zone

Table 80: show security policies hit-count Output Fields *(continued)*

Field Name	Field Description
name	Name of the security policy
policy count	Number of hits for each security policy
tenant	Displays the name of the tenant system.

Sample Output

show security policies hit-count

```
user@host> show security policies hit-count

index   from zone    to zone       name       policy count
 1       untrust      vrtrust      policy1         40
 2       untrust      trust        policy2         20
 3       untrust      trust        policy3         80

Number of policy: 3
```

Sample Output

show security policies hit-count ascending

```
user@host> show security policies hit-count ascending

index   from zone    to zone       name       policy count
 2       untrust      trust        policy2         20
 1       untrust      vrtrust      policy1         40
 3       untrust      trust        policy3         80

Number of policy: 3
```

Sample Output

show security policies hit-count descending greater-than 70 less-than 100

```
user@host> show security policies hit-count descending greater-than 70 less-than 100
```

```
index   from zone    to zone       name       policy count
 2       untrust      trust        policy2         100
 1       untrust      vrtrust      policy1         90
 3       untrust      trust        policy3         80

Number of policy: 3
```

Sample Output

show security policies hit-count from-zone untrust to-zone trust

```
user@host> show security policies hit-count from-zone untrust to-zone trust
```

```
index   from zone    to zone       name       policy count
 2       untrust      trust        policy2         20
 3       untrust      trust        policy3         80

Number of policy: 2
```

Sample Output

show security policies hit-count tenant all

```
user@host> show security policies hit-count tenant all
```

```
Tenant: TN1
Index   From zone        To zone          Name          Policy count
1       trust            untrust          p12           14
2       trust            untrust          p11           0

Number of policy: 2
```

show security resource-manager group active

Syntax

```
show security resource-manager group active
<group-number >
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5; **node** options added in Junos OS Release 9.0.

Description

Display security information about active groups created through the resource manager.

Options

- none—Display resource manager group service information for all active groups.
- ***group-number*** —(Optional) Display resource manager group service information for a specific group identification number.
- **node**—(Optional) For chassis cluster configurations, display active resource manager group service information on a specific node.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security resource-manager group active on page 764
show security resource-manager group active 2048 on page 764
show security resource-manager group active node primary on page 765
show security resource-manager group active node all on page 765
show security resource-manager group active 1024 node all on page 765

Output Fields

Table 81 on page 764 lists the output fields for the **show security resource-manager group** command. Output fields are listed in the approximate order in which they appear.

Table 81: show security resource-manager group Output Fields

Field Name	Field Description
Total groups	Total number of groups in the system.
active groups	Number of active groups.
Group ID	Identification number whose group information is displayed.

Sample Output

show security resource-manager group active

```
user@host> show security resource-manager group active
```

```
Total groups 32, active groups 0
```

Sample Output

show security resource-manager group active 2048

```
user@host> show security resource-manager group active 2048
```

```
Total groups 2048, active groups 1
Group ID 2048: state - Active
          : Virtual System      - root
          : Application         - SIP ALG
          : Group Timeout       - 65535
          : Number of resources - 3
               Resource ID - 8190
               Resource ID - 8188
               Resource ID - 8187
```

Sample Output

show security resource-manager group active node primary

user@host> **show security resource-manager group active node primary**

```
node0:
--------------------------------------------------------------------------
Group ID 1024: Application - SIP ALG
Total groups 1024, active groups 1
```

Sample Output

show security resource-manager group active node all

user@host> **show security resource-manager group active node all**

```
node0:
--------------------------------------------------------------------------
Group ID 1024: Application - SIP ALG
Total groups 1024, active groups 1
node1:
--------------------------------------------------------------------------
Group ID 1024: Application - SIP ALG
Total groups 1024, active groups 1
```

Sample Output

show security resource-manager group active 1024 node all

user@host> **show security resource-manager group active 1024 node all**

```
node0:
--------------------------------------------------------------------------
Group ID 1024: state - Active
          : Application         - SIP ALG
          : Group Timeout       - 65535
          : Number of resources - 3
               Resource ID - 8192
```

```
               Resource ID - 8188
               Resource ID - 8187
node1:
--------------------------------------------------------------------------------
Group ID 1024: state - Active
          : Application         - SIP ALG
          : Group Timeout       - 65535
          : Number of resources - 3
               Resource ID - 8187
               Resource ID - 8186
               Resource ID - 8190
```

show security resource-manager resource active

Syntax

```
show security resource-manager resource active
<resource-id >
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5; **node** options added in Junos OS Release 9.0.

Description

Display security information about active resources created through the resource manager.

Options

- none—Display information for all active resources.
- ***resource-id*** —(Optional) Display information for a resource with a specific identification number.
- **node**—(Optional) For chassis cluster configurations, display active resource manager information on a specific node.
  - ***node-id*** —Identification number of the node. It can be 0 or 1.
  - **all**—Display information about all nodes.
  - **local**—Display information about the local node.
  - **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security resource-manager resource active on page 768
show security resource-manager resource active 5 on page 768
show security resource-manager resource active node local on page 769
show security resource-manager resource active node primary on page 769

Output Fields

Table 82 on page 768 lists the output fields for the **show security resource-manager resource** command. Output fields are listed in the approximate order in which they appear.

Table 82: show security resource-manager resource Output Fields

Field Name	Field Description
Total resources	Total number of resources in the system.
active resources	Number of active resources.
Resource ID	Identification number whose resource information is displayed.

Sample Output

show security resource-manager resource active

```
user@host> show security resource-manager resource active
```

```
    Resource ID 7: Group ID - 2, Application - JSF_sip

    Resource ID 6: Group ID - 2, Application - JSF_sip

    Resource ID 5: Group ID - 2, Application - JSF_sip

    Resource ID 4: Group ID - 2, Application - JSF_sip

    Resource ID 3: Group ID - 2, Application - JSF_sip

    Resource ID 1: Group ID - 2, Application - JSF_sip

    Resource ID 2: Group ID - 2, Application - JSF_sip
Total Resources 4326, active resources 7
```

Sample Output

show security resource-manager resource active 5

```
user@host> show security resource-manager resource active 5
```

```
Resource ID 5: state - Active
    Application         - asl_client
    Parent group        - 2
    Policy              - 5
    From zone           - untrust
    To zone             - trust
    Resource timeout    - 0
    Number of sessions  - 0
    Number of Holes     - 1
        Source IP range        - {0.0.0.0, 0.0.0.0}
        Source port range      - {0, 0}
        Destination IP range   - {33.1.0.200, 33.1.0.200}
        Destination port range - {5060, 5060}
        Translated             - {0.0.0.0/0 -> 33.1.0.200/5060}
        Protocol               - 17
        Reference count        - 1
```

Sample Output

show security resource-manager resource active node local

`user@host>` **show security resource-manager resource active node local**

```
node0:
--------------------------------------------------------------------------
    Resource ID 8192: Group ID - 1024, Application - SIP ALG
    Resource ID 8188: Group ID - 1024, Application - SIP ALG
    Resource ID 8187: Group ID - 1024, Application - SIP ALG
Total Resources 8192, active resources 3
```

Sample Output

show security resource-manager resource active node primary

`user@host>` **show security resource-manager resource active node primary**

```
node0:
--------------------------------------------------------------------------
    Resource ID 8192: Group ID - 1024, Application - SIP ALG
```

```
    Resource ID 8188: Group ID - 1024, Application - SIP ALG
    Resource ID 8187: Group ID - 1024, Application - SIP ALG
Total Resources 8192, active resources 3
```

show security resource-manager settings

Syntax

```
show security resource-manager settings
<node ( node-id  | all | local | primary)>
```

Release Information

Command introduced in Junos OS Release 8.5; **node** options added in Junos OS Release 9.0.

Description

Display resource manager settings.

Options

node—(Optional) For chassis cluster configurations, display resource manager settings on a specific node.

- ***node-id*** —Identification number of the node. It can be 0 or 1.
- **all**—Display information about all nodes.
- **local**—Display information about the local node.
- **primary**—Display information about the primary node.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security resource-manager settings on page 772
show security resource-manager settings node primary on page 772
show security resource-manager settings node all on page 772

Output Fields

Table 83 on page 772 lists the output fields for the **show security resource-manager settings** command. Output fields are listed in the approximate order in which they appear.

Table 83: show security resource-manager settings Output Fields

Field Name	Field Description
Client Heartbeat	Time after which idle an resource manager client is timed out.
Count	Number of active clients.
Pinhole age	Duration for which the temporary opening in the security firewall (pinhole) is open for specified traffic. If the specified traffic does not exist during this time period, the pinhole is timed out.

Sample Output

show security resource-manager settings

```
user@host> show security resource-manager settings
```

```
Client Heartbeat: timeout 600 seconds, count 5
Pinhole age: 32 seconds
```

Sample Output

show security resource-manager settings node primary

```
user@host> show security resource-manager settings node primary
```

```
node0:
--------------------------------------------------------------------------
Client heartbeat: timeout 600 seconds, count 5
Pinhole age: 120 seconds
```

Sample Output

show security resource-manager settings node all

```
user@host> show security resource-manager settings node all
```

```
node0:
--------------------------------------------------------------------------------
Client heartbeat: timeout 600 seconds, count 5
Pinhole age: 120 seconds
node1:
--------------------------------------------------------------------------------
Client heartbeat: timeout 600 seconds, count 5
Pinhole age: 120 seconds
```

show security resource-manager summary

Syntax

```
show security resource-manager summary
```

Release Information

Command introduced in Junos OS Release 11.4.

Description

Display summary information about active resources, clients, groups, and sessions created through the resource manager.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

List of Sample Output

show security resource-manager summary on page 775

Output Fields

Table 84 on page 774 lists the output fields for the **show security resource-manager summary** command. Output fields are listed in the approximate order in which they appear.

Table 84: show security resource-manager summary Output Fields

Field Name	Field Description
Active resource-manager clients	Number of active resource manager clients.
Active resource-manager groups	Number of active resource manager groups.
Active resource-manager resources	Number of active resource manager resources.
Active resource-manager sessions	Number of active resource manager sessions.

Sample Output

show security resource-manager summary

```
user@host> show security resource-manager summary
```

```
Active resource-manager clients   : 15
Active resource-manager groups    : 1
Active resource-manager resources : 1
Active resource-manager sessions  : 0
```

show security screen ids-option

Syntax

```
show security screen ids-option
screen-name
logical-system
root-logical-system
tenant
```

Release Information

Command introduced in Junos OS Release 8.5. Support for UDP **port scan** added in Junos OS Release 12.1X47-D10.
Support for **node** option added in Junos OS Release 9.0.
Support for IPv6 extension header screens added in Junos OS Release 12.1X46-D10.
The **tenant** option is introduced in Junos OS Release 18.3R1.

Description

Display the configuration information about the specified security screen. You can configure a **ids-option** to enable screen protection on the SRX Series devices.

Options

- ***screen-name*** —Name of the screen.
- **logical-system**—Name of the logical system.
- **root-logical-system**—Displays root logical system as default.
- **tenant**—Name of the tenant system.

Required Privilege Level

view

RELATED DOCUMENTATION

> *ids-option*
>
> *Example: Configuring Multiple Screening Options*

List of Sample Output

show security screen ids-option jscreen on page 779
show security screen ids-option jscreen (IPv6) on page 780
show security screen ids-option jscreen1 node all on page 781
show security screen ids-option jscreen tenant TN1 on page 781

show security screen ids-option jscreen tenant all on page 782

Output Fields

Table 85 on page 777 lists the output fields for the **show security screen ids-option** command. Output fields are listed in the approximate order in which they appear.

Table 85: show security screen ids-option Output Fields

Field Name	Field Description
TCP address sweep threshold	Number of microseconds for which the device accepts 10 TCP packets from the same remote source to different destination addresses.
TCP port scan threshold	Number of microseconds during which the device accepts packets from the same remote source with up to 10 different port numbers.
ICMP address sweep threshold	Number of microseconds during which up to 10 ICMP echo requests from the same host are allowed into the device.
UDP flood threshold	Number of UDP packets per second allowed to ping the same destination address before the device rejects further UDP packets.
UDP port scan threshold	Number of microseconds during which the device accepts packets from the same remote source IP with up to 10 different destination port numbers.
TCP winnuke	Enable or disable the detection of TCP WinNuke attacks.
TCP SYN flood attack threshold	Number of SYN packets per second required to trigger the SYN proxy response.
TCP SYN flood alarm threshold	Number of half-complete proxy connections per second at which the device makes entries in the event alarm log.
TCP SYN flood source threshold	Number of SYN segments to be received per second before the device begins dropping connection requests.
TCP SYN flood destination threshold	Number of SYN segments received per second before the device begins dropping connection requests.
TCP SYN flood timeout	Maximum length of time before a half-completed connection is dropped from the queue.
TCP SYN flood queue size	Number of proxy connection requests that can be held in the proxy connection queue before the device begins rejecting new connection requests.
ICMP large packet	Enable or disable the detection of any ICMP frame with an IP length greater than 1024 bytes.

Table 85: show security screen ids-option Output Fields *(continued)*

Field Name	Field Description
UDP address sweep threshold	Number of microseconds for which the device accepts 10 UDP packets from the same remote source to different destination addresses.
IPv6 extension routing	Enable or disable the IPv6 extension routing screen option.
IPv6 extension shim6	Enable or disable the IPv6 extension shim6 screen option.
IPv6 extension fragment/IP block fragment	Enable or disable the IPv6 extension fragment screen option.
IPv6 extension AH	Enable or disable the IPv6 extension Authentication Header Protocol screen option.
IPv6 extension ESP	Enable or disable the IPv6 extension Encapsulating Security Payload screen option.
IPv6 extension mobility	Enable or disable the IPv6 extension mobility screen option.
IPv6 extension HIP	Enable or disable the IPv6 extension Host Identify Protocol screen option.
IPv6 extension no next	Enable or disable the IPv6 extension no-next screen option.
IPv6 extension user-defined	Enable or disable the IPv6 extension user-defined screen option.
IPv6 extension HbyH jumbo	Enable or disable the IPv6 extension HbyH jumbo screen option.
IPv6 extension HbyH RPL	Enable or disable the IPv6 extension HbyH RPL screen option.
IPv6 extension HbyH router alert	Enable or disable the IPv6 extension HbyH router screen option.
IPv6 extension HbyH quick start	Enable or disable the IPv6 extension HbyH quick-start screen option.
IPv6 extension HbyH CALIPSO	Enable or disable the IPv6 extension HbyH Common Architecture Label IPv6 Security Screen option.
IPv6 extension HbyH SMF DPD	Enable or disable the IPv6 extension HbyH Simplified Multicast Forwarding IPv6 Duplicate Packet Detection screen option.
IPv6 extension HbyH user-defined	Enable or disable the IPv6 extension HbyH user-defined screen option.
IPv6 extension Dst tunnel encap limit	Enable or disable the IPv6 extension distributed (network) storage tunnel encapsulation limit screen option.

Table 85: show security screen ids-option Output Fields *(continued)*

Field Name	Field Description
IPv6 extension Dst home address	Enable or disable the IPv6 extension DST home address screen option.
IPv6 extension Dst ILNP nonce	Enable or disable the IPv6 extension DST Identifier-Locator Network Protocol nonce screen option.
IPv6 extension Dst line-id	Enable or disable the IPv6 extension DST line-ID screen option.
IPv6 extension Dst user-defined	Enable or disable the IPv6 extension DST user-defined screen option.
IPv6 extension header limit	Threshold for the number of IPv6 extension headers that can pass through the screen.
IPv6 malformed header	Enable or disable the IPv6 malformed header screen option.
ICMPv6 malformed header	Enable or disable the ICMPv6 malformed packet screen option.
UDP flood white-list	Whitelist of IP addresses to bypass UDP flood detection.
Session source limit threshold	Limit the number of concurrent sessions the device can initiate from a single source IP address or the number of sessions it can direct to a single destination IP address.
Logical system/Tenant	Name of the logical system or tenant system.

Sample Output

show security screen ids-option jscreen

`user@host>` **show security screen ids-option jscreen**

```
Screen object status:
Name                                    Value
  TCP port scan threshold               5000
  UDP port scan threshold               10000
  ICMP address sweep threshold          5000
```

Sample Output

show security screen ids-option jscreen (IPv6)

user@host> **show security screen ids-option jscreen**

```
Screen object status:

   Name                                        Value
  ICMP ping of death                          enabled
  ……

  IPv6 extension routing                      enabled
  IPv6 extension shim6                        enabled
  IPv6 extension fragment                     enabled
  IPv6 extension AH                           enabled
  IPv6 extension ESP                          enabled
  IPv6 extension mobility                     enabled
  IPv6 extension HIP                          enabled
  IPv6 extension no next                      enabled
  IPv6 extension user-defined                 enabled
  IPv6 extension HbyH jumbo                   enabled
  IPv6 extension HbyH RPL                     enabled
  IPv6 extension HbyH router alert            enabled
  IPv6 extension HbyH quick start             enabled
  IPv6 extension HbyH CALIPSO                 enabled
  IPv6 extension HbyH SMF DPD                 enabled
  IPv6 extension HbyH user-defined            enabled
  IPv6 extension Dst tunnel encap limit       enabled
  IPv6 extension Dst home address             enabled
  IPv6 extension Dst ILNP nonce               enabled
  IPv6 extension Dst line-id                  enabled
  IPv6 extension Dst user-defined             enabled
  IPv6 extension header limit                 20
  IPv6 Malformed header                       enabled
  ICMPv6 malformed packet                     enabled
```

Sample Output

show security screen ids-option jscreen1 node all

```
user@host> show security screen ids-option jscreen1 node all
```

```
node0:
--------------------------------------------------------------------------
Screen object status:
Name                                        Value
  UDP flood threshold                       1000
  TCP winnuke                               enabled
  TCP SYN flood attack threshold            200
  TCP SYN flood alarm threshold             512
  TCP SYN flood source threshold            4000
  TCP SYN flood destination threshold       4000
  TCP SYN flood timeout                     20
  TCP SYN flood queue size                  1024
  ICMP large packet                         enabled
node1:
--------------------------------------------------------------------------
Screen object status:
Name                                        Value
  UDP flood threshold                       1000
  TCP winnuke                               enabled
  TCP SYN flood attack threshold            200
  TCP SYN flood alarm threshold             512
  TCP SYN flood source threshold            4000
  TCP SYN flood destination threshold       4000
  TCP SYN flood timeout                     20
  TCP SYN flood queue size                  1024
  ICMP large packet                         enabled
```

show security screen ids-option jscreen tenant TN1

```
user@host> show security screen ids-option jscreen tenant TN1
```

```


Screen object status:

Name                                      value
UDP flood threshold                       1000
UDP flood white-list                      a1
```

```
UDP flood white-list                        a2
```

show security screen ids-option jscreen tenant all

user@host> **show security screen ids-option jscreen tenant all**

```
Logical system: root-logical-system
Screen object status:

Name                                        value
UDP flood threshold                         1
UDP flood white-list                        a1
UDP flood white-list                        a2
IP block fragment                           enabled
Session source limit threshold              5

Tenant: TN1
Screen object status:

Name                                        value
UDP flood threshold                         1000
UDP flood white-list                        a1
UDP flood white-list                        a2
```

show security screen statistics

Syntax

```
show security screen statistics <zone zone-name | interface interface-name>
logical-system <logical-system-name | all>
root-logical-system
tenant <tenant-name >
```

Release Information

Command introduced in Junos OS Release 8.5.
The **node** option added in Junos OS Release 9.0.
The **logical-system all** option added in Junos OS Release 11.2R6.
Support for IPv6 extension header screens added in Junos OS Release 12.1X46-D10.
The **tenant** option is introduced in Junos OS Release 18.3R1.

Description

Display intrusion detection service (IDS) security screen statistics.

Options

- **zone *zone-name***—Display screen statistics for this security zone.
- **interface *interface-name*** —Display screen statistics for this interface.
- ***logical-system-name***—Display screen statistics for the named logical system.
- **root-logical-system**—(Optional) Display screen statistics for the master logical system only.
- **tenant**—Display the name of the tenant system.

Required Privilege Level

view

RELATED DOCUMENTATION

clear security screen statistics

clear security screen statistics interface

clear security screen statistics zone

Example: Configuring Multiple Screening Options

List of Sample Output

show security screen statistics zone scrzone on page 786
show security screen statistics zone untrust (IPv6) on page 787

show security screen statistics interface ge-0/0/3 on page 788
show security screen statistics interface ge-0/0/1 (IPv6) on page 789
show security screen statistics interface ge-0/0/1 node primary on page 789
show security screen statistics zone trust logical-system all on page 790
show security screen statistics zone trust tenant TN1 on page 793
show security screen statistics zone trust tenant all on page 794

Output Fields

Table 86 on page 784 lists the output fields for the **show security screen statistics** command. Output fields are listed in the approximate order in which they appear.

Table 86: show security screen statistics Output Fields

Field Name	Field Description
ICMP flood	Internet Control Message Protocol (ICMP) flood counter. An ICMP flood typically occurs when ICMP echo requests use all resources in responding, such that valid network traffic can no longer be processed.
UDP flood	User Datagram Protocol (UDP) flood counter. UDP flooding occurs when an attacker sends IP packets containing UDP datagrams with the purpose of slowing down the resources, such that valid connections can no longer be handled.
TCP winnuke	Number of Transport Control Protocol (TCP) WinNuke attacks. WinNuke is a denial-of-service (DoS) attack targeting any computer on the Internet running Windows.
TCP port scan	Number of TCP port scans. The purpose of this attack is to scan the available services in the hopes that at least one port will respond, thus identifying a service to target.
ICMP address sweep	Number of ICMP address sweeps. An IP address sweep can occur with the intent of triggering responses from active hosts.
IP tear drop	Number of teardrop attacks. Teardrop attacks exploit the reassembly of fragmented IP packets.
TCP SYN flood	Number of TCP SYN attacks.
IP spoofing	Number of IP spoofs. IP spoofing occurs when an invalid source address is inserted in the packet header to make the packet appear to come from a trusted source.
ICMP ping of death	ICMP ping of death counter. Ping of death occurs when IP packets are sent that exceed the maximum legal length (65,535 bytes).

Table 86: show security screen statistics Output Fields *(continued)*

Field Name	Field Description
IP source route option	Number of IP source route attacks.
TCP address sweep	Number of TCP address sweeps.
TCP land attack	Number of land attacks. Land attacks occur when an attacker sends spoofed SYN packets containing the IP address of the victim as both the destination and source IP address.
TCP SYN fragment	Number of TCP SYN fragments.
TCP no flag	Number of TCP headers without flags set. A normal TCP segment header has at least one control flag set.
IP unknown protocol	Number of IPs.
IP bad options	Number of invalid options.
IP record route option	Number of packets with the IP record route option enabled. This option records the IP addresses of the network devices along the path that the IP packet travels.
IP timestamp option	Number of IP timestamp option attacks. This option records the time (in Universal Time) when each network device receives the packet during its trip from the point of origin to its destination.
IP security option	Number of IP security option attacks.
IP loose source route option	Number of IP loose source route option attacks. This option specifies a partial route list for a packet to take on its journey from source to destination.
IP strict source route option	Number of IP strict source route option attacks. This option specifies the complete route list for a packet to take on its journey from source to destination.
IP stream option	Number of stream option attacks. This option provides a way for the 16-bit SATNET stream identifier to be carried through networks that do not support streams.
ICMP fragment	Number of ICMP fragments. Because ICMP packets contain very short messages, there is no legitimate reason for ICMP packets to be fragmented. If an ICMP packet is so large that it must be fragmented, something is amiss.
ICMP large packet	Number of large ICMP packets.

Table 86: show security screen statistics Output Fields *(continued)*

Field Name	Field Description
TCP SYN FIN	Number of TCP SYN FIN packets.
TCP FIN no ACK	Number of TCP FIN flags without the acknowledge (ACK) flag.
Source session limit	Number of concurrent sessions that can be initiated from a source IP address.
TCP SYN-ACK-ACK proxy	Number of TCP flags enabled with SYN-ACK-ACK. To prevent flooding with SYN-ACK-ACK sessions, you can enable the SYN-ACK-ACK proxy protection screen option. After the number of connections from the same IP address reaches the SYN-ACK-ACK proxy threshold and SRX Series devices running Junos OS reject further connection requests from that IP address.
IP block fragment	Number of IP block fragments.
Destination session limit	Number of concurrent sessions that can be directed to a single destination IP address.
UDP address sweep	Number of UDP address sweeps.
IPv6 extension header	Number of packets filtered for the defined IPv6 extension headers.
IPv6 extension hop by hop option	Number of packets filtered for the defined IPv6 hop-by-hop option types.
IPv6 extension destination option	Number of packets filtered for the defined IPv6 destination option types.
IPv6 extension header limit	Number of packets filtered for crossing the defined IPv6 extension header limit.
IPv6 malformed header	Number of IPv6 malformed headers defined for the intrusion detection service (IDS).
ICMPv6 malformed packet	Number of ICMPv6 malformed packets defined for the IDS options.

Sample Output

show security screen statistics zone scrzone

```
user@host> show security screen statistics zone scrzone
```

```
Screen statistics:
IDS attack type                          Statistics
  ICMP flood                             0
  UDP flood                              0
  TCP winnuke                            0
  TCP port scan                          91
  ICMP address sweep                     0
  TCP sweep                              0
  UDP sweep                              0
  IP tear drop                           0
  TCP SYN flood                          0
  IP spoofing                            0
  ICMP ping of death                     0
  IP source route option                 0
  TCP land attack                        0
  TCP SYN fragment                       0
  TCP no flag                            0
  IP unknown protocol                    0
  IP bad options                         0
  IP record route option                 0
  IP timestamp option                    0
  IP security option                     0
  IP loose source route option           0
  IP strict source route option          0
  IP stream option                       0
  ICMP fragment                          0
  ICMP large packet                      0
  TCP SYN FIN                            0
  TCP FIN no ACK                         0
  Source session limit                   0
  TCP SYN-ACK-ACK proxy                  0
  IP block fragment                      0
  Destination session limit              0
```

Sample Output

show security screen statistics zone untrust (IPv6)

`user@host>` **show security screen statistics zone untrust**

```
Screen statistics:
```

```
IDS attack type                          Statistics
  ICMP flood                                   0
  UDP  flood                                   0
  TCP  winnuke                                 0
  ……
  IPv6   extension header                       0
  IPv6   extension hop by hop option            0
  IPv6   extension destination option           0
  IPv6   extension header limit                 0
  IPv6   malformed header                       0
  ICMPv6 malformed packet                       0
```

Sample Output

show security screen statistics interface ge-0/0/3

`user@host>` **show security screen statistics interface ge-0/0/3**

```
Screen statistics:
IDS attack type                          Statistics
  ICMP flood                             0
  UDP flood                              0
  TCP winnuke                            0
  TCP port scan                          91
  ICMP address sweep                     0
  TCP sweep                              0
  UDP sweep                              0
  IP tear drop                           0
  TCP SYN flood                          0
  IP spoofing                            0
  ICMP ping of death                     0
  IP source route option                 0
  TCP land attack                        0
  TCP SYN fragment                       0
  TCP no flag                            0
  IP unknown protocol                    0
  IP bad options                         0
  IP record route option                 0
  IP timestamp option                    0
  IP security option                     0
  IP loose source route option           0
```

```
  IP strict source route option              0
  IP stream option                           0
  ICMP fragment                              0
  ICMP large packet                          0
  TCP SYN FIN                                0
  TCP FIN no ACK                             0
  Source session limit                       0
  TCP SYN-ACK-ACK proxy                      0
  IP block fragment                          0
  Destination session limit                  0
```

Sample Output

show security screen statistics interface ge-0/0/1 (IPv6)

`user@host>` **show security screen statistics interface ge-0/0/1**

```
Screen statistics:

IDS attack type                                 Statistics
  ICMP flood                                         0
  UDP flood                                          0
  ……
  IPv6 extension header                              0
  IPv6 extension hop by hop option                   0
  IPv6 extension destination option                  0
  IPv6 extension header limit                        0
  IPv6 malformed header                              0
  ICMPv6 malformed packet                            0
```

Sample Output

show security screen statistics interface ge-0/0/1 node primary

`user@host>` **show security screen statistics interface ge-0/0/1 node primary**

```
node0:
--------------------------------------------------------------------------
```

```
Screen statistics:
IDS attack type                              Statistics
  ICMP flood                                 1
  UDP flood                                  1
  TCP winnuke                                1
  TCP port scan                              1
  ICMP address sweep                         1
  TCP sweep                                  1
  UDP sweep                                  1
  IP tear drop                               1
  TCP SYN flood                              1
  IP spoofing                                1
  ICMP ping of death                         1
  IP source route option                     1
  TCP land attack                            1
  TCP SYN fragment                           1
  TCP no flag                                1
  IP unknown protocol                        1
  IP bad options                             1
  IP record route option                     1
  IP timestamp option                        1
  IP security option                         1
  IP loose source route option               1
  IP strict source route option              1
  IP stream option                           1
  ICMP fragment                              1
  ICMP large packet                          1
  TCP SYN FIN                                1
  TCP FIN no ACK                             1
  Source session limit                       1
  TCP SYN-ACK-ACK proxy                      1
  IP block fragment                          1
  Destination session limit                  1
```

Sample Output

show security screen statistics zone trust logical-system all

`user@host>` **show security screen statistics zone trust logical-system all**

```
Logical system: root-logical-system
Screen statistics:
```

```
IDS attack type                          Statistics
  ICMP flood                             0
  UDP flood                              0
  TCP winnuke                            0
  TCP port scan                          0
  ICMP address sweep                     0
  TCP sweep                              0
  UDP sweep                              0
  IP tear drop                           0
  TCP SYN flood                          0
  IP spoofing                            0
  ICMP ping of death                     0
  IP source route option                 0
  TCP land attack                        0
  TCP SYN fragment                       0
  TCP no flag                            0
  IP unknown protocol                    0
  IP bad options                         0
  IP record route option                 0
  IP timestamp option                    0
  IP security option                     0
  IP loose source route option           0
  IP strict source route option          0
  IP stream option                       0
  ICMP fragment                          0
  ICMP large packet                      0
  TCP SYN FIN                            0
  TCP FIN no ACK                         0
  Source session limit                   0
  TCP SYN-ACK-ACK proxy                  0
  IP block fragment                      0
  Destination session limit              0

Logical system: ls1
Screen statistics:

IDS attack type                          Statistics
  ICMP flood                             0
  UDP flood                              0
  TCP winnuke                            0
  TCP port scan                          0
  ICMP address sweep                     0
  TCP sweep                              0
```

```
  UDP sweep                                0
  IP tear drop                             0
  TCP SYN flood                            0
  IP spoofing                              0
  ICMP ping of death                       0
  IP source route option                   0
  TCP land attack                          0
  TCP SYN fragment                         0
  TCP no flag                              0
  IP unknown protocol                      0
  IP bad options                           0
  IP record route option                   0
  IP timestamp option                      0
  IP security option                       0
  IP loose source route option             0
  IP strict source route option            0
  IP stream option                         0
  ICMP fragment                            0
  ICMP large packet                        0
  TCP SYN FIN                              0
  TCP FIN no ACK                           0
  Source session limit                     0
  TCP SYN-ACK-ACK proxy                    0
  IP block fragment                        0
  Destination session limit                0

Logical system: ls2
Screen statistics:

IDS attack type                          Statistics
  ICMP flood                             0
  UDP flood                              0
  TCP winnuke                            0
  TCP port scan                          0
  ICMP address sweep                     0
  TCP sweep                              0
  UDP sweep                              0
  IP tear drop                           0
  TCP SYN flood                          0
  IP spoofing                            0
  ICMP ping of death                     0
  IP source route option                 0
  TCP land attack                        0
  TCP SYN fragment                       0
```

```
TCP no flag                              0
IP unknown protocol                      0
IP bad options                           0
IP record route option                   0
IP timestamp option                      0
IP security option                       0
IP loose source route option             0
IP strict source route option            0
IP stream option                         0
ICMP fragment                            0
ICMP large packet                        0
TCP SYN FIN                              0
TCP FIN no ACK                           0
Source session limit                     0
TCP SYN-ACK-ACK proxy                    0
IP block fragment                        0
Destination session limit                0
```

show security screen statistics zone trust tenant TN1

user@host> **show security screen statistics zone trust tenant TN1**

```
Screen statistics:

IDS attack type                          Statistics
ICMP flood                               0
UDP flood                                0
TCP winnuke                              0
TCP port scan                            0
UDP port scan                            0
ICMP address sweep                       0
TCP sweep                                0
UDP sweep                                0
IP tear drop                             0
TCP SYN flood                            0
SYN flood source                         0
SYN flood destination                    0
IP spoofing                              0
ICMP ping of death                       0
IP source route option                   0
TCP land attack                          0
TCP SYN fragment                         0
TCP no flag                              0
```

```
IP unknown protocol                         0
IP bad options                              0
IP record route option                      0
IP timestamp option                         0
IP security option                          0
IP loose source route option                0
IP strict source route option               0
IP stream option                            0
ICMP fragment                               0
ICMP large packet                           0
TCP SYN FIN                                 0
TCP FIN no ACK                              0
Source session limit                        0
TCP SYN-ACK-ACK proxy                       0
IP block fragment                           0
Destination session limit                   0
IPv6 extension header                       0
IPv6 extension hop by hop option            0
IPv6 extension destination option           0
IPv6 extension header limit                 0
IPv6 malformed header                       0
ICMPv6 malformed packet                     0
IP tunnel summary                           0
```

show security screen statistics zone trust tenant all

user@host> **show security screen statistics zone trust tenant all**

```
Logical system: root-logical-system
creen statistics:

IDS attack type                             Statistics
ICMP flood                                  0
UDP flood                                   0
TCP winnuke                                 0
TCP port scan                               0
UDP port scan                               0
ICMP address sweep                          0
TCP sweep                                   0
UDP sweep                                   0
IP tear drop                                0
TCP SYN flood                               0
SYN flood source                            0
```

```
SYN flood destination                    0
IP spoofing                              0
ICMP ping of death                       0
IP source route option                   0
TCP land attack                          0
TCP SYN fragment                         0
TCP no flag                              0
IP unknown protocol                      0
IP bad options                           0
IP record route option                   0
IP timestamp option                      0
IP security option                       0
IP loose source route option             0
IP strict source route option            0
IP stream option                         0
ICMP fragment                            0
ICMP large packet                        0
TCP SYN FIN                              0
TCP FIN no ACK                           0
Source session limit                     0
TCP SYN-ACK-ACK proxy                    0
IP block fragment                        0
Destination session limit                0
IPv6 extension header                    0
IPv6 extension hop by hop option         0
IPv6 extension destination option        0
IPv6 extension header limit              0
IPv6 malformed header                    0
ICMPv6 malformed packet                  0
IP tunnel summary                        0

Tenant: TN1
Screen statistics:

IDS attack type                          Statistics
ICMP flood                               0
UDP flood                                0
TCP winnuke                              0
TCP port scan                            0
UDP port scan                            0
ICMP address sweep                       0
TCP sweep                                0
UDP sweep                                0
IP tear drop                             0
```

```
TCP SYN flood                          0
SYN flood source                       0
SYN flood destination                  0
IP spoofing                            0
ICMP ping of death                     0
IP source route option                 0
TCP land attack                        0
TCP SYN fragment                       0
TCP no flag                            0
IP unknown protocol                    0
IP bad options                         0
IP record route option                 0
IP timestamp option                    0
IP security option                     0
IP loose source route option           0
IP strict source route option          0
IP stream option                       0
ICMP fragment                          0
ICMP large packet                      0
TCP SYN FIN                            0
TCP FIN no ACK                         0
Source session limit                   0
TCP SYN-ACK-ACK proxy                  0
IP block fragment                      0
Destination session limit              0
IPv6 extension header                  0
IPv6 extension hop by hop option       0
IPv6 extension destination option      0
IPv6 extension header limit            0
IPv6 malformed header                  0
ICMPv6 malformed packet                0
IP tunnel summary                      0
```

show security softwires

Syntax

```
show security softwires <softwire-name softwire-name>
<logical-system (all | logical-system-name)>
```

Release Information

Command introduced in Junos OS Release 10.4. The **logical-system** option introduced in Junos OS Release 12.1.

Description

Display a summary of information of all the softwire concentrators and details on concentrators with specified name.

Options

softwire-name *softwire-name*—Display the details of the specified softwire concentrator.

logical-system (all | *logical-system-name*)—Display softwire information for all logical systems or for a specified logical system. This option is only available to the master administrator.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Traffic Processing on Security Devices | 25

Sample Output

`user@host>` **show security softwires**

```
Softwire Name           SC Address         Status   Number of SI connected
  SC-CSSI-1                3001::1          Connected      2
  SC-CSSI-str00            3100::1          Active         0
  SC-CSSI-str01            3101::1          Inactive       0
  SC-CSSI-str02            3001::1          Connected      2520
```

user@host> **show security softwires softwire-name SC-CSSI-1**

```
Name of softwire: SC-CSSI-1
   SC status: Connected
   SC address: 3001::1
   Zone: trust
   VR ID: 0
 SI Address               SI Status                  SPU
 3001::2                  Active                     spu-1
 3001::2                  Active                     spu-21
 SI number: 2
```

user@host> **show security softwires logical-system ls-product-design**

```
Softwire Name            SC Address         Status   Number of SI connected
  sc_1                      3000::1         Connected      1
```

show security zones

Syntax

```
show security zones
<zone-name>
<all-logical-systems-tenants>
<detail>
<logical-system (logical-system-name | all)>
<root-logical-system>
<tenant (tenant-name | all)>
<terse>
<type (functional | security)>
```

Release Information

Command introduced in Junos OS Release 8.5.
tenant option introduced in Junos OS Release 18.3R1.

Description

Displays the information about the security zones. You can define a security zone, which allows you to divide the network into different segments and apply different security options to each segment.

Options

- ***none***—Displays information about all the security zones configured.
- ***zone-name***—(Optional) Displays information about the specified security zone.
- **all-logical-systems-tenants**—(Optional) Displays the information about the security zone of all logical systems and tenant systems.
- **detail**—(Optional) Displays the detail level information about the security zone.
- **logical-system *logical-system-name***—(Optional) Displays the information about the security zones of a specified logical system.
- **logical-system all**—(Optional) Displays the information about the security zones of all logical systems.
- **root-logical-system**—(Optional) Displays the information about the security zones of the root logical system.
- **tenant *tenant-name***—(Optional) Displays the information about the security zones of a specified tenant system.
- **tenant all**—(Optional) Displays the information about the security zones of all tenant systems.
- **terse**—(Optional) Displays the specified level information about the security zone.

- **type functional**—(Optional) Displays the information for functional zones.
- **type security**—(Optional) Displays the information for security zones.

Required Privilege Level
view

RELATED DOCUMENTATION

> *Security Zones Overview*
>
> *Supported System Services for Host Inbound Traffic*
>
> *security-zone*

List of Sample Output
show security zones on page 801
show security zones abc on page 802
show security zones all-logical-systems-tenants on page 802
show security zones abc detail on page 803
show security zones logical-system LSYS1 on page 804
show security zones logical-system all on page 804
show security zones root-logical-system on page 804
show security zones tenant TSYS1 on page 805
show security zone tenant all on page 805
show security zones terse on page 806
show security zones type security on page 806

Output Fields
Table 87 on page 800 lists the output fields for the **show security zones** command. Output fields are listed in the approximate order in which they appear.

Table 87: show security zones Output Fields

Field Name	Field Description	Level of Output
Functional zone	Name of the functional zone.	none
Security zone	Name of the security zone.	detail none
Description	Description of the security zone.	detail none

Table 87: show security zones Output Fields *(continued)*

Field Name	Field Description	Level of Output
Policy configurable	Whether the policy can be configured or not.	detail none
Interfaces bound	Number of interfaces in the zone.	detail none
Interfaces	List of the interfaces in the zone.	detail none
Zone	Name of the zone.	terse
Type	Type of the zone.	terse
Logical system	Name of the logical system.	detail
Tenant	Name of the tenant system.	detail

Sample Output

show security zones

`user@host>` **show security zones**

```
Functional zone: management
   Description: This is the management zone.
   Policy configurable: No
   Interfaces bound: 1
   Interfaces:
     ge-0/0/0.0
Security zone: Host
   Description: This is the host zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     fxp0.0
Security zone: abc
```

```
   Description: This is the abc zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/1.0
Security zone: def
   Description: This is the def zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/2.0
```

show security zones abc

user@host> **show security zones abc**

```
Security zone: abc
   Description: This is the abc zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/1.0
```

show security zones all-logical-systems-tenants

user@host> **show security zones all-logical-systems-tenants**

```
Logical system: root-logical-system

Security zone: HOST
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: all
  Interfaces:

Security zone: junos-host
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
```

```
  Interfaces:

Logical system: LSYS1

Security zone: z1
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

Logical system: TSYS1

Security zone: z3a
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

Security zone: z3b
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

Security zone: z3c
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

show security zones abc detail

user@host> **show security zones abc detail**

```
Security zone: abc
   Description: This is the abc zone.
   Send reset for non-SYN session TCP packets: Off
   Policy configurable: Yes
   Interfaces bound: 1
   Interfaces:
     ge-0/0/1.0
```

show security zones logical-system LSYS1

```
user@host> show security zones logical-system LSYS1
```

```
Security zone: z1
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

show security zones logical-system all

```
user@host> show security zones logical-system all
```

```
Logical system: root-logical-system

Security zone: HOST
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: all
  Interfaces:

Security zone: junos-host
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

Logical system: LSYS1

Security zone: z1
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

show security zones root-logical-system

```
user@host> show security zones root-logical-system
```

```
Security zone: HOST
```

```
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: all
  Interfaces:

Security zone: junos-host
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

show security zones tenant TSYS1

user@host> **show security zones tenant TSYS1**

```
Security zone: z3a
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

Security zone: z3b
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

Security zone: z3c
 Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

show security zone tenant all

user@host> **show security zone tenant all**

```
  Tenant: TSYS1


  Security zone: Host
```

```
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:

  Security zone: abc
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:xe-0/0/1.0

  Security zone: def
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 1
  Interfaces:xe-0/0/3.0
```

show security zones terse

user@host> **show security zones terse**

```
Zone                    Type
my-internal             Security
my-external             Security
dmz                     Security
```

show security zones type security

user@host> **show security zones type security**

```
Security zone: HOST
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: all
  Interfaces:

Security zone: junos-host
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

show security zones type

Syntax

```
show security zones type
(functional | security)
<detail | terse>
```

Release Information

Command introduced in Junos OS Release 8.5. The **Description** output field added in Junos OS Release 12.1.

Description

Display information about security zones of the specified type.

Options

- **functional**—Display functional zones.
- **security**—Display security zones.
- **detail** | **terse**—(Optional) Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

> *Security Zones Overview*
>
> *Supported System Services for Host Inbound Traffic*
>
> *security-zone*

List of Sample Output

show security zones type functional on page 808
show security zones type security on page 809
show security zones type security terse on page 809
show security zones type security detail on page 810

Output Fields

Table 88 on page 808 lists the output fields for the **show security zones type** command. Output fields are listed in the approximate order in which they appear.

Table 88: show security zones type Output Fields

Field Name	Field Description	Level of Output
Security zone	Zone name.	All levels
Description	Description of the security zone.	none detail
Policy configurable	Whether the policy can be configured or not.	none detail
Interfaces bound	Number of interfaces in the zone.	none detail
Interfaces	List of the interfaces in the zone.	none detail
Zone	Name of the zone.	All levels
Type	Type of the zone.	All levels

Sample Output

show security zones type functional

`user@host>` **show security zones type functional**

```
Functional zone: management
  Description: management zone
  Policy configurable: No
  Interfaces bound: 0
  Interfaces:
```

Sample Output

show security zones type security

user@host> **show security zones type security**

```
Security zone: trust
  Description: trust zone
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 1
  Interfaces:
  ge-0/0/0.0
Security zone: untrust
  Description: untrust zone
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 1
  Interfaces:
  ge-0/0/1.0
Security zone: junos-host
  Description: junos-host zone
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```

Sample Output

show security zones type security terse

user@host> **show security zones type security terse**

```
Zone                    Type
trust                   Security
untrust                 Security
junos-host              Security
```

Sample Output

show security zones type security detail

user@host> **show security zones type security detail**

```
Security zone: trust
  Description: trust zone
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 1
  Interfaces:
  ge-0/0/0.0
Security zone: untrust
  Description: untrust zone
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 1
  Interfaces:
  ge-0/0/1.0
Security zone: junos-host
  Description: junos-host zone
  Send reset for non-SYN session TCP packets: Off
  Policy configurable: Yes
  Interfaces bound: 0
  Interfaces:
```