



---

Junos<sup>®</sup> OS

## Flow-Based and Packet-Based Processing Feature Guide for Security Devices



---

Modified: 2017-11-22

Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, California 94089  
USA  
408-745-2000  
[www.juniper.net](http://www.juniper.net)

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. and/or its affiliates in the United States and other countries. All other trademarks may be 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 Feature Guide for Security Devices*  
Copyright © 2017 Juniper Networks, Inc. All rights reserved.

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

#### YEAR 2000 NOTICE

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

#### END USER LICENSE AGREEMENT

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

# Table of Contents

	About the Documentation . . . . .	xix
	Documentation and Release Notes . . . . .	xix
	Supported Platforms . . . . .	xix
	Using the Examples in This Manual . . . . .	xix
	Merging a Full Example . . . . .	xx
	Merging a Snippet . . . . .	xx
	Documentation Conventions . . . . .	xxi
	Documentation Feedback . . . . .	xxiii
	Requesting Technical Support . . . . .	xxiii
	Self-Help Online Tools and Resources . . . . .	xxiii
	Opening a Case with JTAC . . . . .	xxiv
<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Introduction to Processing on Security Devices . . . . .</b>	<b>3</b>
	Juniper Networks Devices Processing Overview . . . . .	3
	Understanding Flow-Based Processing . . . . .	4
	Zones and Policies . . . . .	5
	Flows and Sessions . . . . .	5
	Understanding Packet-Based Processing . . . . .	5
	Stateless Firewall Filters . . . . .	6
	Class-of-Service Features . . . . .	6
	Screens . . . . .	6
	Understanding SRX Series Services Gateways Central Point Architecture . . . . .	7
	Load Distribution in Combo Mode . . . . .	8
	Sharing Processing Power and Memory in Combo Mode . . . . .	9
	Understanding Enhancements to Central Point Architecture for the SRX5000 Line . . . . .	9
	SRX5000 Line Devices Processing Overview . . . . .	11
	Understanding First-Packet Processing . . . . .	12
	Understanding Fast-Path Processing . . . . .	13
	Understanding the Data Path for Unicast Sessions . . . . .	14
	Session Lookup and Packet-Match Criteria . . . . .	14
	Understanding Session Creation: First-Packet Processing . . . . .	15
	Understanding Fast-Path Processing . . . . .	17
	Understanding Services Processing Units . . . . .	20
	Understanding Scheduler Characteristics . . . . .	20
	Understanding Network Processor Bundling . . . . .	21
	Network Processor Bundling Limitations . . . . .	21
	Understanding Central Point Session Limit Performance Enhancements . . . . .	21

SRX3000 Line and SRX1400 Devices Processing Overview . . . . .	22
Components Involved in Setting Up a Session . . . . .	23
Understanding the Data Path for Unicast Sessions . . . . .	24
Session Lookup and Packet Match Criteria . . . . .	24
Understanding Session Creation: First Packet Processing . . . . .	24
Understanding Fast-Path Processing . . . . .	26
SRX210 and SRX320 Services Gateway Processing Overview . . . . .	26
Understanding Flow Processing and Session Management . . . . .	27
Understanding First-Packet Processing . . . . .	27
Understanding Session Creation . . . . .	27
Understanding Fast-Path Processing . . . . .	28
Understanding Central Point Architecture Flow Support for GTP and SCTP . . . .	28
Understanding the Flow Session Connection Filter Option . . . . .	29

## Part 2

### Chapter 2

## Configuring Flow-Based Sessions

<b>Configuring Security Flow Sessions . . . . .</b>	<b>33</b>
Understanding Session Characteristics for SRX Series Services Gateways . . . .	33
Understanding Aggressive Session Aging . . . . .	34
Example: Controlling Session Termination for SRX Series Services Gateways . .	35
Understanding TCP Session Checks per Policy . . . . .	37
Disabling TCP Packet Security Checks . . . . .	38
Example: Disabling TCP Packet Security Checks for SRX Series Services	
Gateways . . . . .	38
Example: Setting the Maximum Segment Size for All TCP Sessions for SRX	
Series Services Gateways . . . . .	39
Example: Configuring TCP Packet Security Checks Per Policy . . . . .	41
TCP Out-of-State Packet Drop Logging Overview . . . . .	42
Understanding TCP Out-of-State Packet Drop Logging . . . . .	44
Supported TCP Out-of-State Logging Features . . . . .	44
Packet Filter Component . . . . .	44
Throttle Component . . . . .	44
Flexibility for Changing the Log Generation Rate . . . . .	45
Example: Configuring TCP Out-of-State Packet Logging on SRX Series	
Devices . . . . .	45
Configuring the Timeout Value for Multicast Flow Sessions . . . . .	48
Clearing Sessions for SRX Series Services Gateways . . . . .	50
Terminating Sessions for SRX Series Services Gateways . . . . .	50
Terminating a Specific Session for SRX Series Services Gateways . . . . .	50
Using Filters to Specify the Sessions to Be Terminated for SRX Series	
Services Gateways . . . . .	50
Understanding the Default Processing Behavior for IPv4 Traffic . . . . .	50
Understanding How Preserving Incoming Fragmentation Characteristics Can	
Improve Throughput . . . . .	52

	Understanding ECMP Flow-Based Forwarding . . . . .	53
	Understanding ECMP Flow-Based Forwarding for Reverse Traffic on SRX Series Devices and vSRX Instances . . . . .	55
	ECMP Overview . . . . .	55
	ECMP Implementation for Junos OS SRX Series Devices and vSRX Instances . . . . .	55
	ECMP for Reverse Traffic . . . . .	56
	Example: Configuring ECMP Flow-Based Forwarding . . . . .	57
	Understanding ECMP Flow-Based Forwarding . . . . .	57
	Example: Configuring ECMP Flow-Based Forwarding . . . . .	58
<b>Part 3</b>	<b>Improving Flow-Based Performance</b>	
<b>Chapter 3</b>	<b>Expanding Session Capacity by Device . . . . .</b>	<b>67</b>
	Expanding Session Capacity by Device . . . . .	67
	Expanding Session Capacity on an SRX3400 or SRX3600 Device . . . . .	68
	Reverting to Default Session Capacity on an SRX5800 Device . . . . .	68
	Verifying the Current Session Capacity . . . . .	68
<b>Chapter 4</b>	<b>Managing Sessions and Flow Distribution . . . . .</b>	<b>71</b>
	Understanding Load Distribution in SRX5800, SRX5600, and SRX5400 Devices . . . . .	71
	Hash-Based Forwarding on the SRX5K-MPC, SRX5K-MPC3-40G10G (IOC3), and the SRX5K-MPC3-100G10G (IOC3) . . . . .	72
	Understanding Packet-Ordering Function on SRX5000 Line Devices . . . . .	73
	Disabling Packet-Ordering Mode on SRX5000 Line Devices . . . . .	74
	Understanding Session Distribution on High End SRX Series Devices in Adaptive Mode . . . . .	75
<b>Chapter 5</b>	<b>Reducing Long Packet-Processing Latency by Express Path . . . . .</b>	<b>79</b>
	Understanding Session Cache . . . . .	79
	Overview . . . . .	79
	Selective Session Cache Installation . . . . .	81
	IPsec VPN Session Affinity Enhancement Using Session Cache . . . . .	82
	Fragmentation Packet Ordering Using NP Session Cache . . . . .	82
	Express Path Overview . . . . .	83
	Understanding Express Path Functionality . . . . .	84
	Understanding Express Path Support on SRX Series Devices . . . . .	85
	Understanding Express Path Features . . . . .	86
	Wing Statistics Counter . . . . .	86
	Cross-Network Traffic . . . . .	88
	LAG Support in Express Path Mode . . . . .	88
	End-to-End Debugging . . . . .	90
	Per-Session Statistics CLI . . . . .	90
	Express Path Limitations . . . . .	91
	Express Path Support on NP-IOC Card . . . . .	92
	Express Path Support on SRX5K Modular Port Concentrator . . . . .	93
	Express Path Support on SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) . . . . .	94
	IPv6 Flow in Express Path Mode for IOC2 and IOC3 . . . . .	96

	IPv6 Flow in Express Path Mode . . . . .	97
	Understanding the Services Offload Solution . . . . .	99
	Enabling and Disabling Express Path . . . . .	100
	Example: Enabling Express Path in Security Policies . . . . .	102
	Example: Configuring an NPC on SRX3000 Line Devices or SRX1400 Devices to Support Express Path . . . . .	103
	Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path . . . . .	105
	Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path . . . . .	107
	Example: Configuring SRX5K-MPC3-100G10G (IOC3) and SRX5K-MPC3-40G10G (IOC3) on an SRX5000 Line Device to Support Express Path . . . . .	109
	Example: Configuring Express Path on an SRX5000 Line Device with IOC3 for IPv6 Traffic . . . . .	112
	Example: Configuring Low Latency . . . . .	117
<b>Chapter 6</b>	<b>Managing IPsec Tunnel Fragmentation . . . . .</b>	<b>121</b>
	Understanding the Fragmentation Counters Feature . . . . .	121
	Fragmentation Counters Feature Overview . . . . .	121
	Understanding Fragmentation and MTU and MSS Sizes . . . . .	122
	Using Fragmentation Counter Statistics to Tune Your System . . . . .	123
<b>Part 4</b>	<b>Managing Flow-Based Processing for IPv6</b>	
<b>Chapter 7</b>	<b>Enabling IPv6 Flow-Based Processing . . . . .</b>	<b>127</b>
	IPv6 Advanced Flow . . . . .	127
	Understanding Sessions for IPv6 Flows . . . . .	129
	Understanding IPv6 Flow Processing on SRX5400, SRX5600, and SRX5800 devices . . . . .	129
	Flow-Based Processing for IPv6 Traffic . . . . .	132
	Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways . . . . .	134
<b>Chapter 8</b>	<b>Managing IPv6 Packets . . . . .</b>	<b>141</b>
	The IPv6 Packet Header and SRX Series Overview . . . . .	141
	About the IPv6 Basic Packet Header . . . . .	142
	Understanding IPv6 Packet Header Extensions . . . . .	144
	About IPv6 Packet Header Verification Performed by the Flow Module for SRX Series Devices . . . . .	145
	Understanding Path MTU Messages for IPv6 Packets . . . . .	145
	Understanding How SRX Series Devices Handle Packet Fragmentation for IPv6 Flows . . . . .	147
	Understanding How SRX Series Devices Handle ICMPv6 Packets . . . . .	147
<b>Chapter 9</b>	<b>Configuring IPv6 Dual-Stack . . . . .</b>	<b>151</b>
	Understanding IPv6 Dual-Stack Lite . . . . .	151
	Example: Configuring IPv6 Dual-Stack Lite . . . . .	154

<b>Part 5</b>	<b>Monitoring Flow-Based Sessions and Establishing Parameters for Error Handling</b>	
<b>Chapter 10</b>	<b>Monitoring Security Flow Sessions</b>	<b>161</b>
	Monitoring Security Flow Sessions Overview	161
	Understanding How to Obtain Session Information for SRX Series Services Gateways	162
	Displaying Global Session Parameters for All SRX Series Services Gateways	164
	Displaying a Summary of Sessions for SRX Series Services Gateways	165
	Displaying Session and Flow Information About Sessions for SRX Series Services Gateways	166
	Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways	166
	Using Filters to Display Session and Flow Information for SRX Series Services Gateways	167
	Information Provided in Session Log Entries for SRX Series Services Gateways	167
	Error Handling Extensions	172
	Understanding Chassis Manager FPC Fault Detection and Error Handling Enhancements	173
	Error Handling on IOCs and SPCs	173
	Error Detection and Management	173
	Error Detection Processes	174
	Integration with Chassis Cluster	175
	Wedge Detection, Reporting, and Management	175
<b>Chapter 11</b>	<b>Monitoring X2 Traffic By Configuring Mirror Filters</b>	<b>177</b>
	Understanding X2 Traffic Monitoring	177
	X2 Traffic Monitoring Overview	177
	Limitations of X2 Traffic Monitoring	179
	X2 Traffic Terminology	179
	Example: Configuring a Mirror Filter for X2 Traffic Monitoring	180
<b>Part 6</b>	<b>Configuring Packet-Based Forwarding</b>	
<b>Chapter 12</b>	<b>Configuring Selective Stateless Packet-Based Services</b>	<b>187</b>
	Understanding Packet-Based Processing	187
	Understanding Selective Stateless Packet-Based Services	188
	Selective Stateless Packet-Based Services Configuration Overview	190
	Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding	192
	Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding	203
<b>Part 7</b>	<b>Configuration Statements and Operational Commands</b>	
<b>Chapter 13</b>	<b>Configuration Statements</b>	<b>217</b>
	aging	219
	all-tcp	220
	allow-dns-reply	220

allow-embedded-icmp	221
allow-reverse-ecmp	221
application-services (Security Forwarding Process)	222
apply-to-half-close-state	223
ethernet-switching	224
destination-header	225
destination-port (Security Forwarding Options)	226
destination-prefix (Security Forwarding Options)	230
early-ageout	230
error	231
fin-invalidate-session	234
flow (Security Flow)	236
force-ip-reassembly	238
forwarding-process	239
fpc error	241
fru-poweron-sequence	243
gre-in	244
gre-out	245
high-watermark	246
hop-by-hop-header	247
icmpv6-malformed	248
idle-timeout (System Services)	249
inline-tap	250
interface-in (Security Forwarding Options)	250
interface-out (Security Forwarding Options)	251
ipv4-template (Services)	251
ipv6-extension-header	252
ipv6-extension-header-limit	253
ipv6-malformed-header	254
ipv6-template (Services)	254
low-latency	255
low-watermark	256
maximize-idp-sessions	257
mirror-filter (Security Forwarding Options)	258
mode (Security Forwarding Options)	260
no-sequence-check	261
no-tcp-reset	262
output (Security Forwarding Options)	263
packet-filter	264
packet-log (Security Flow)	265
packet-ordering-mode (Application Services)	266
pending-sess-queue-length	267
per-session-statistics	267
preserve-incoming-fragment-size	268
propagate-settings	269
protocol (Security Forwarding Options)	270
resource-manager	271
route-change-timeout	272
rst-invalidate-session	273



	rst-sequence-check . . . . .	274
	sampling . . . . .	275
	services-offload . . . . .	276
	session (System Services) . . . . .	277
	session-limit (System Services) . . . . .	278
	np-cache (Flexible PIC Concentrator) . . . . .	279
	source-port (Security Forwarding Options) . . . . .	280
	source-prefix (Security Forwarding Options) . . . . .	284
	syn-flood-protection-mode . . . . .	284
	tcp-initial-timeout . . . . .	285
	tcp-mss (Security Flow) . . . . .	286
	tcp-session . . . . .	287
	time-wait-state . . . . .	288
	traceoptions (Security) . . . . .	289
	traceoptions (Security Flow) . . . . .	291
	transport (Security Log) . . . . .	295
	weight (Security) . . . . .	296
<b>Chapter 14</b>	<b>Operational Commands . . . . .</b>	<b>297</b>
	clear firewall . . . . .	300
	clear monitor security flow filter . . . . .	301
	clear security flow ip-action . . . . .	302
	clear security flow session all . . . . .	304
	clear security flow session application . . . . .	306
	clear security flow session destination-port . . . . .	308
	clear security flow session family . . . . .	310
	clear security flow session interface . . . . .	311
	clear security flow session protocol . . . . .	313
	clear security flow session resource-manager . . . . .	315
	clear security flow session services-offload . . . . .	317
	clear security flow session session-identifier . . . . .	320
	clear security flow session source-port . . . . .	322
	clear security flow session source-prefix . . . . .	324
	clear security forward-options mirror filter . . . . .	326
	monitor security flow file . . . . .	327
	monitor security flow filter . . . . .	329
	monitor security flow start . . . . .	331
	monitor security flow stop . . . . .	332
	show chassis environment (Security) . . . . .	333
	show chassis fpc (View) . . . . .	337
	show chassis fpc errors . . . . .	345
	show chassis hardware (View) . . . . .	347
	show chassis pic (Security) . . . . .	358
	show chassis power . . . . .	360
	show chassis power sequence . . . . .	363
	show firewall (View) . . . . .	364
	show interfaces (View Aggregated Ethernet) . . . . .	366
	show interfaces (SRX Series) . . . . .	377
	show interfaces diagnostics optics . . . . .	409

show interfaces flow-statistics . . . . .	414
show interfaces swfabx . . . . .	419
show monitor security flow . . . . .	421
show security flow cp-session . . . . .	423
show security flow cp-session destination-port . . . . .	426
show security flow cp-session destination-prefix . . . . .	429
show security flow cp-session family . . . . .	432
show security flow cp-session protocol . . . . .	435
show security flow cp-session source-port . . . . .	438
show security flow cp-session source-prefix . . . . .	441
show security flow gate . . . . .	444
show security flow ip-action . . . . .	448
show security flow gate brief node . . . . .	456
show security flow gate destination-port . . . . .	462
show security flow gate destination-prefix . . . . .	465
show security flow gate protocol . . . . .	468
show security flow gate summary node . . . . .	471
show security flow session . . . . .	476
show security flow session brief node . . . . .	483
show security flow session destination-port . . . . .	487
show security flow session destination-prefix . . . . .	491
show security flow session extensive node . . . . .	495
show security flow session family . . . . .	501
show security flow session interface . . . . .	506
show security flow session nat . . . . .	510
show security flow session policy-id . . . . .	513
show security flow session protocol . . . . .	516
show security flow session resource-manager . . . . .	521
show security flow session services-offload . . . . .	525
show security flow session session-identifier . . . . .	530
show security flow session source-port . . . . .	535
show security flow session source-prefix . . . . .	539
show security flow session summary family . . . . .	543
show security flow session summary node . . . . .	545
show security flow session summary services-offload . . . . .	551
show security flow session tunnel . . . . .	555
show security flow statistics . . . . .	564
show security flow status . . . . .	570
show security forwarding-options mirror-filter . . . . .	573
show security monitoring . . . . .	576
show security policies . . . . .	578
show security policies hit-count . . . . .	587
show security resource-manager group active . . . . .	590
show security resource-manager resource active . . . . .	593
show security resource-manager settings . . . . .	596
show security resource-manager summary . . . . .	598
show security screen ids-option . . . . .	599
show security screen statistics . . . . .	604
show security softwires . . . . .	612

show security zones . . . . .	613
show security zones type . . . . .	616



# List of Figures

<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Introduction to Processing on Security Devices</b>	<b>3</b>
	Figure 1: Traffic Flow for Flow-Based Processing	4
	Figure 2: First-Packet Processing	12
	Figure 3: Fast-Path Processing	13
	Figure 4: Session Creation: First-Packet Processing	17
	Figure 5: Packet Walk for Fast-Path Processing	20
<b>Part 2</b>	<b>Configuring Flow-Based Sessions</b>	
<b>Chapter 2</b>	<b>Configuring Security Flow Sessions</b>	<b>33</b>
	Figure 6: ECMP Routes	59
<b>Part 3</b>	<b>Improving Flow-Based Performance</b>	
<b>Chapter 5</b>	<b>Reducing Long Packet-Processing Latency by Express Path</b>	<b>79</b>
	Figure 7: IOC3 Intra-PFE Express Path	95
	Figure 8: IOC3 Inter-PFE Express Path	96
	Figure 9: Inter-IOC3 Express Path	96
<b>Part 4</b>	<b>Managing Flow-Based Processing for IPv6</b>	
<b>Chapter 9</b>	<b>Configuring IPv6 Dual-Stack</b>	<b>151</b>
	Figure 10: DS-Lite NAT (IPv4-in-IPv6)	152
<b>Part 5</b>	<b>Monitoring Flow-Based Sessions and Establishing Parameters for Error Handling</b>	
<b>Chapter 11</b>	<b>Monitoring X2 Traffic By Configuring Mirror Filters</b>	<b>177</b>
	Figure 11: SRX Series Device in an LTE Mobile Network	178
	Figure 12: Monitoring X2 Traffic	178
	Figure 13: Configuring Mirror Filters for X2 Traffic Monitoring	181
<b>Part 6</b>	<b>Configuring Packet-Based Forwarding</b>	
<b>Chapter 12</b>	<b>Configuring Selective Stateless Packet-Based Services</b>	<b>187</b>
	Figure 14: Traffic Flow for Packet-Based Forwarding	187
	Figure 15: Traffic Flow with Selective Stateless Packet-Based Services	189
	Figure 16: Intranet Traffic Using End-to-End Packet-Based Services	194
	Figure 17: Selective Stateless Packet-Based Services for Packet-Based Forwarding	205



# List of Tables

	<b>About the Documentation</b> . . . . .	<b>xix</b>
	Table 1: Notice Icons . . . . .	xxi
	Table 2: Text and Syntax Conventions . . . . .	xxii
<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Introduction to Processing on Security Devices</b> . . . . .	<b>3</b>
	Table 3: Central Point Implementation on SRX Series Devices in Conjunction With Junos OS Releases . . . . .	7
	Table 4: Combo Mode Processing . . . . .	9
<b>Part 2</b>	<b>Configuring Flow-Based Sessions</b>	
<b>Chapter 2</b>	<b>Configuring Security Flow Sessions</b> . . . . .	<b>33</b>
	Table 5: How the Final Egress MTU Size for Fragments Exiting the SRX Series Device Is Determined . . . . .	53
<b>Part 3</b>	<b>Improving Flow-Based Performance</b>	
<b>Chapter 3</b>	<b>Expanding Session Capacity by Device</b> . . . . .	<b>67</b>
	Table 6: Maximum Central Point Session Increases . . . . .	67
<b>Chapter 5</b>	<b>Reducing Long Packet-Processing Latency by Express Path</b> . . . . .	<b>79</b>
	Table 7: Session Cache Installation Bars . . . . .	81
	Table 8: Session Cache Table Utilization Bits Status . . . . .	82
	Table 9: Express Path Support on SRX Series Device Cards . . . . .	85
	Table 10: Total Number of Sessions per Wing in Network Processor Express Path Configuration Mode . . . . .	87
<b>Part 4</b>	<b>Managing Flow-Based Processing for IPv6</b>	
<b>Chapter 8</b>	<b>Managing IPv6 Packets</b> . . . . .	<b>141</b>
	Table 11: IPv6 Extension Headers . . . . .	144
	Table 12: ICMPv6 Packets That Junos OS Handles Differently from Other ICMPv6 Packets . . . . .	148
<b>Chapter 9</b>	<b>Configuring IPv6 Dual-Stack</b> . . . . .	<b>151</b>
	Table 13: Software Initiator and Software Concentrator Capacity . . . . .	153
<b>Part 5</b>	<b>Monitoring Flow-Based Sessions and Establishing Parameters for Error Handling</b>	
<b>Chapter 10</b>	<b>Monitoring Security Flow Sessions</b> . . . . .	<b>161</b>

	Table 14: Session Create Log Fields . . . . .	168
	Table 15: Session Close Log Fields . . . . .	169
	Table 16: Session Deny Log Fields . . . . .	171
<b>Chapter 11</b>	<b>Monitoring X2 Traffic By Configuring Mirror Filters . . . . .</b>	<b>177</b>
	Table 17: X2 Traffic Terminology . . . . .	179
<b>Part 7</b>	<b>Configuration Statements and Operational Commands</b>	
<b>Chapter 13</b>	<b>Configuration Statements . . . . .</b>	<b>217</b>
	Table 18: Ports Supported by Services Interfaces . . . . .	227
	Table 19: Value Ranges for Error Levels . . . . .	232
	Table 20: Terminating a Session with a 4-Way Handshake . . . . .	235
	Table 21: Device Status Upon Configuration Change . . . . .	261
	Table 22: Ports Supported by Services Interfaces . . . . .	281
	Table 23: Session Capacity and Resulting Throughput . . . . .	296
<b>Chapter 14</b>	<b>Operational Commands . . . . .</b>	<b>297</b>
	Table 24: show chassis environment Output Fields . . . . .	333
	Table 25: SRX5K-MPC3-40G10G (IOC3) PIC Selection Summary . . . . .	337
	Table 26: show chassis fpc Output Fields . . . . .	339
	Table 27: show chassis fpc errors Output Fields . . . . .	345
	Table 28: show chassis hardware Output Fields . . . . .	347
	Table 29: show chassis pic Output Fields . . . . .	358
	Table 30: show chassis power Output Fields . . . . .	360
	Table 31: show chassis power sequence Output Fields . . . . .	363
	Table 32: show firewall Output Fields . . . . .	364
	Table 33: Aggregated Ethernet show interfaces Output Fields . . . . .	367
	Table 34: show interfaces Output Fields . . . . .	380
	Table 35: show interfaces diagnostics optics Output Fields . . . . .	409
	Table 36: show interfaces flow-statistics Output Fields . . . . .	415
	Table 37: Flow Error Statistics (Packet Drop Statistics for the Flow Module) . . . . .	415
	Table 38: show interfaces <swfab0   swfab1> Output Fields . . . . .	419
	Table 39: show monitor security flow Output Fields . . . . .	421
	Table 40: show security flow cp-session Output Fields . . . . .	424
	Table 41: show security flow cp-session destination-port Output Fields . . . . .	426
	Table 42: show security flow cp-session destination-prefix Output Fields . . . . .	429
	Table 43: show security flow cp-session family Output Fields . . . . .	432
	Table 44: show security flow cp-session protocol Output Fields . . . . .	436
	Table 45: show security flow cp-session source-port Output Fields . . . . .	438
	Table 46: show security flow cp-session source-prefix Output Fields . . . . .	441
	Table 47: show security flow gate Output Fields . . . . .	445
	Table 48: show security flow ip-action Output Fields . . . . .	449
	Table 49: show security flow gate brief node Output Fields . . . . .	456
	Table 50: show security flow gate destination-port Output Fields . . . . .	462
	Table 51: show security flow gate destination-prefix Output Fields . . . . .	465
	Table 52: show security flow gate protocol Output Fields . . . . .	469
	Table 53: show security flow gate summary node Output Fields . . . . .	471
	Table 54: show security flow session Output Fields . . . . .	478
	Table 55: show security flow session brief node Output Fields . . . . .	483



Table 56: show security flow session destination-port Output Fields . . . . .	487
Table 57: show security flow session destination-prefix Output Fields . . . . .	491
Table 58: show security flow session extensive node Output Fields . . . . .	495
Table 59: show security flow session family Output Fields . . . . .	501
Table 60: show security flow session interface Output Fields . . . . .	506
Table 61: show security flow session nat Output Fields . . . . .	510
Table 62: show security flow session policy-id Output Fields . . . . .	513
Table 63: show security flow session protocol Output Fields . . . . .	517
Table 64: show security flow session resource-manager Output Fields . . . . .	521
Table 65: show security flow session services-offload Output Fields . . . . .	526
Table 66: show security flow session session-identifier Output Fields . . . . .	530
Table 67: Tunnel Type Identification . . . . .	532
Table 68: show security flow session source-port Output Fields . . . . .	535
Table 69: show security flow session source-prefix Output Fields . . . . .	539
Table 70: show security flow session summary Output Fields . . . . .	543
Table 71: show security flow session summary node Output Fields . . . . .	545
Table 72: show security flow session summary services-offload Output Fields . . . . .	552
Table 73: show security flow session tunnel Output Fields . . . . .	555
Table 74: show security flow statistics Output Fields . . . . .	565
Table 75: show security flow status Output Fields . . . . .	570
Table 76: show security forward-options mirror-filter . . . . .	574
Table 77: show security policies Output Fields . . . . .	579
Table 78: show security policies hit-count Output Fields . . . . .	588
Table 79: show security resource-manager group Output Fields . . . . .	590
Table 80: show security resource-manager resource Output Fields . . . . .	593
Table 81: show security resource-manager settings Output Fields . . . . .	596
Table 82: show security resource-manager summary Output Fields . . . . .	598
Table 83: show security screen ids-option Output Fields . . . . .	599
Table 84: show security screen statistics Output Fields . . . . .	605
Table 85: show security zones Output Fields . . . . .	613
Table 86: show security zones type Output Fields . . . . .	616



# About the Documentation

- Documentation and Release Notes on page xix
- Supported Platforms on page xix
- Using the Examples in This Manual on page xix
- Documentation Conventions on page xxi
- Documentation Feedback on page xxiii
- Requesting Technical Support on page xxiii

## Documentation and Release Notes

---

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

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

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

## Supported Platforms

---

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

- SRX Series
- vSRX

## 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.xml;
    }
  }
}
interfaces {
  fxp0 {
    disable;
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
    }
  }
}
```

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

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

## Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

For more information about the **load** command, see [CLI Explorer](#).

## Documentation Conventions

Table 1 on page xxi 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 xxii defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
<b>Bold text like this</b>	Represents text that you type.	To enter configuration mode, type the <b>configure</b> command:  user@host> <b>configure</b>
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> <b>show chassis alarms</b>  No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> <li>Introduces or emphasizes important new terms.</li> <li>Identifies guide names.</li> <li>Identifies RFC and Internet draft titles.</li> </ul>	<ul style="list-style-type: none"> <li>A policy <i>term</i> is a named structure that defines match conditions and actions.</li> <li><i>Junos OS CLI User Guide</i></li> <li>RFC 1997, <i>BGP Communities Attribute</i></li> </ul>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name:  [edit] root@# <b>set system domain-name</b> <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"> <li>To configure a stub area, include the <b>stub</b> statement at the [edit protocols ospf area area-id] hierarchy level.</li> <li>The console port is labeled <b>CONSOLE</b>.</li> </ul>
< > (angle brackets)	Encloses optional keywords or variables.	<b>stub</b> <default-metric <i>metric</i> >;
(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.	<b>broadcast</b>   <b>multicast</b>  ( <i>string1</i>   <i>string2</i>   <i>string3</i> )
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	<b>rsvp { # Required for dynamic MPLS only</b>
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	<b>community name members</b> [ <b>community-ids</b> ]
Indentation and braces ( { } )	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	

#### GUI Conventions

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

Convention	Description	Examples
<b>Bold text like this</b>	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none"> <li>In the Logical Interfaces box, select <b>All Interfaces</b>.</li> <li>To cancel the configuration, click <b>Cancel</b>.</li> </ul>
<b>&gt;</b> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select <b>Protocols&gt;Ospf</b> .

## Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page of the Juniper Networks TechLibrary site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <http://www.juniper.net/techpubs/feedback/>.
- E-mail—Send your comments to [techpubs-comments@juniper.net](mailto: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 J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

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

## Self-Help Online Tools and Resources

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

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

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

## Opening a Case with JTAC

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

- Use the Case Management tool in the CSC at <http://www.juniper.net/cm/>.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <http://www.juniper.net/support/requesting-support.html>.



## PART 1

# Overview

- [Introduction to Processing on Security Devices on page 3](#)



## CHAPTER 1

# Introduction to Processing on Security Devices

- [Juniper Networks Devices Processing Overview on page 3](#)
- [Understanding SRX Series Services Gateways Central Point Architecture on page 7](#)
- [Understanding Enhancements to Central Point Architecture for the SRX5000 Line on page 9](#)
- [SRX5000 Line Devices Processing Overview on page 11](#)
- [Understanding Central Point Session Limit Performance Enhancements on page 21](#)
- [SRX3000 Line and SRX1400 Devices Processing Overview on page 22](#)
- [SRX210 and SRX320 Services Gateway Processing Overview on page 26](#)
- [Understanding Central Point Architecture Flow Support for GTP and SCTP on page 28](#)
- [Understanding the Flow Session Connection Filter Option on page 29](#)

## Juniper Networks Devices Processing Overview

---

### Supported Platforms [SRX Series, vSRX](#)

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:

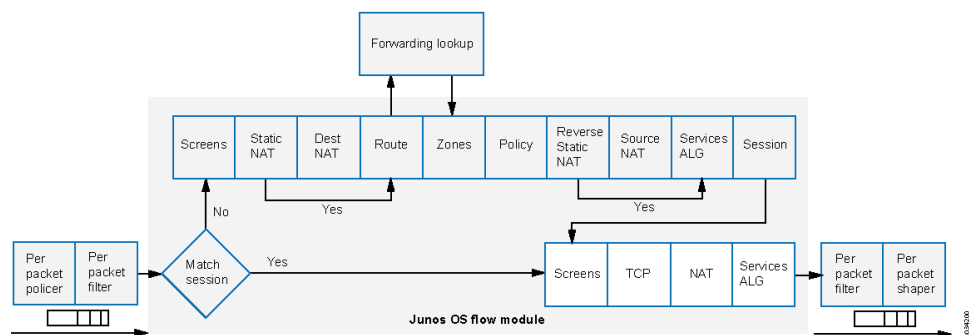
- [Understanding Flow-Based Processing on page 4](#)
- [Understanding Packet-Based Processing on page 5](#)

## 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 4](#) 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 SPU.

---

### Class-of-Service Features

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

- 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).

For details on specific stateless CoS features, see *Class of Service Feature Guide for Security Devices*.

- Related Documentation**
- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)
  - [SRX5000 Line Devices Processing Overview on page 11](#)
  - [SRX210 and SRX320 Services Gateway Processing Overview on page 26](#)

## Understanding SRX Series Services Gateways Central Point Architecture

### Supported Platforms [SRX Series, vSRX](#)

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 combo 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 3 on page 7](#) identifies the central point architecture implementation that is supported on SRX Series devices for various releases.

**Table 3: Central Point Implementation on SRX Series Devices in Conjunction With Junos OS Releases**

	Mode Supported on SRX1400 and SRX1500 Devices	Mode Supported On SRX3000 Series Devices	Mode Supported on SRX5000 Series Devices
Junos OS Release 12.3X48 and Previous Releases	<ul style="list-style-type: none"> <li>• CP centric</li> <li>• combo mode (SRX Series 1400 devices only.)</li> </ul>	<ul style="list-style-type: none"> <li>• CP centric</li> <li>• combo mode</li> </ul>	<ul style="list-style-type: none"> <li>• CP centric</li> <li>• combo mode</li> </ul>

**Table 3: Central Point Implementation on SRX Series Devices in Conjunction With Junos OS Releases (*continued*)**

	Mode Supported on SRX1400 and SRX1500 Devices	Mode Supported On SRX3000 Series Devices	Mode Supported on SRX5000 Series Devices
<ul style="list-style-type: none"> <li>Junos OS Release 15.1X49-D10</li> <li>Junos OS Release 15.1X49-D15</li> <li>Junos OS Release 15.1X49-D20</li> </ul>	These SRX Series devices are no longer supported.	These SRX Series devices are no longer supported.	<ul style="list-style-type: none"> <li>CP-centric</li> </ul> <p><b>NOTE:</b> NG-SPC renders combo mode obsolete.</p>
Junos OS Release 15.1X49-D30 and Following Releases	These SRX Series devices are no longer supported.	These SRX Series devices are no longer supported.	<ul style="list-style-type: none"> <li>CP-lite</li> </ul> <p><b>NOTE:</b> NG-SPC renders combo mode obsolete.</p>

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 *combo*, 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 combo 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 Combo Mode on page 8](#)
- [Sharing Processing Power and Memory in Combo Mode on page 9](#)

## Load Distribution in Combo 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 combo 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 Combo Mode

The CPU processing power in a combo-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 combo mode, CPU functionality takes a small portion of the cores, whereas "medium" SPU combo 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 4 on page 9](#). Platform support depends on the Junos OS release in your installation.

**Table 4: Combo Mode Processing**

SRX Series device	Central point mode with 1 SPC	Central point mode with 2 or More than 2 SPCs
SRX1400	Small	Medium
SRX3400	Small	Medium
SRX3600	Small	Medium
SRX3400 (expanded performance and capacity license)	Small	Large
SRX3600 (expanded performance and capacity license)	Small	Large
SRX5600	Large	Large
SRX5800	Large	Large



**NOTE:** The combo mode processing only exists with SPC1 on SRX1400, SRX3400, SRX3600, and SRX5000 line devices.

### Related Documentation

- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)

## Understanding Enhancements to Central Point Architecture for the SRX5000 Line

**Supported Platforms** [SRX Series, vSRX](#)

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 offloading session management functionalities 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 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.

---

## Release History Table

Release	Description
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

- [SRX5000 Line Devices Processing Overview on page 11](#)

## SRX5000 Line Devices Processing Overview

### Supported Platforms [SRX Series, vSRX](#)

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 SRX5000 line 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. A Network Processing Unit (NPU) runs on an IOC. An IOC has one or more NPUs. 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. Otherwise, however, the responsibilities of these NPUs are not clearly delineated in regard to the other kind of services that run on them. (For details on flow-based processing, see [“Juniper Networks Devices Processing Overview” on page 3.](#))

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:

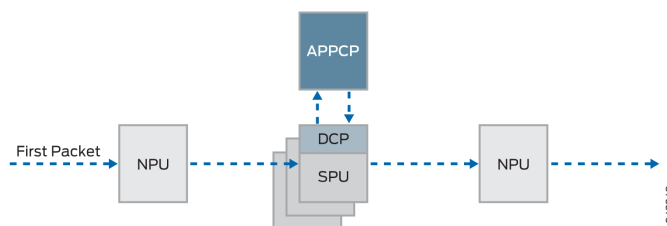
- [Understanding First-Packet Processing on page 12](#)
- [Understanding Fast-Path Processing on page 13](#)
- [Understanding the Data Path for Unicast Sessions on page 14](#)
- [Understanding Services Processing Units on page 20](#)
- [Understanding Scheduler Characteristics on page 20](#)
- [Understanding Network Processor Bundling on page 21](#)

## 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 2 on page 12 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 2: First-Packet Processing**



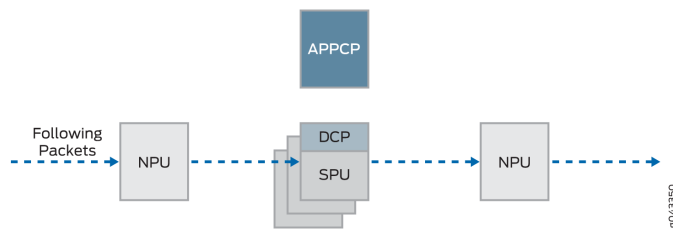
For details on session creation for the first packet in a flow, see [“Understanding Session Creation: First-Packet Processing”](#) on page 15.

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 3 on page 13](#) 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 3: 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. (See [“Understanding Fast-Path Processing” on page 17.](#))

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

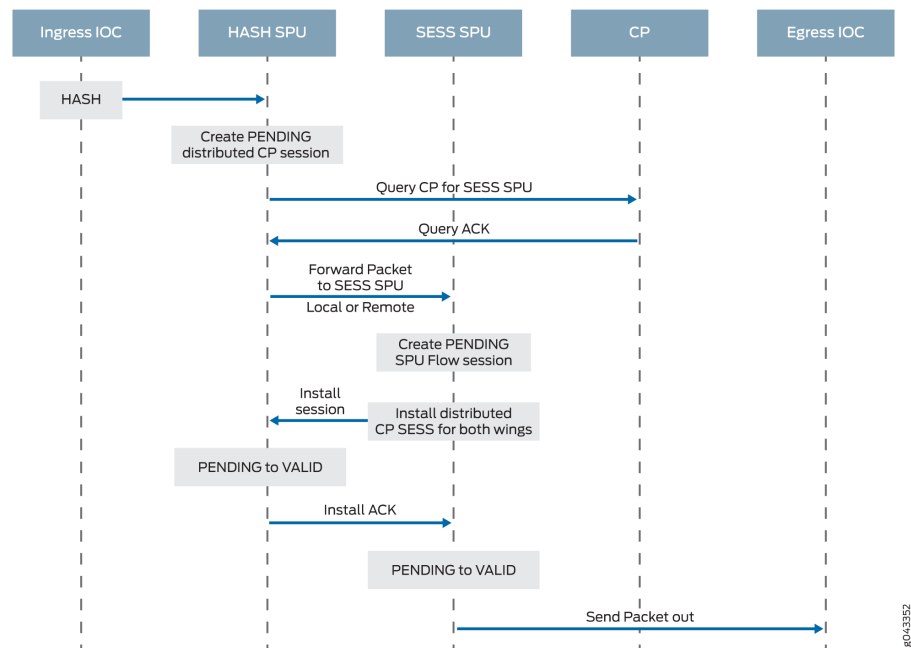
NPUs maintain information about a session for packet forwarding and delivery. Session information is set up on the egress and ingress NPUs (which sometimes are the same) so that packets can be sent directly to the SPU that manages their flows and not to the distributed central point for redirection.

**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 17](#).

[Figure 4 on page 17](#) 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 4: 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, see ["Juniper Networks Devices Processing Overview" on page 3.](#))

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. (See ["Juniper Networks Devices Processing Overview" on page 3.](#))

***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.

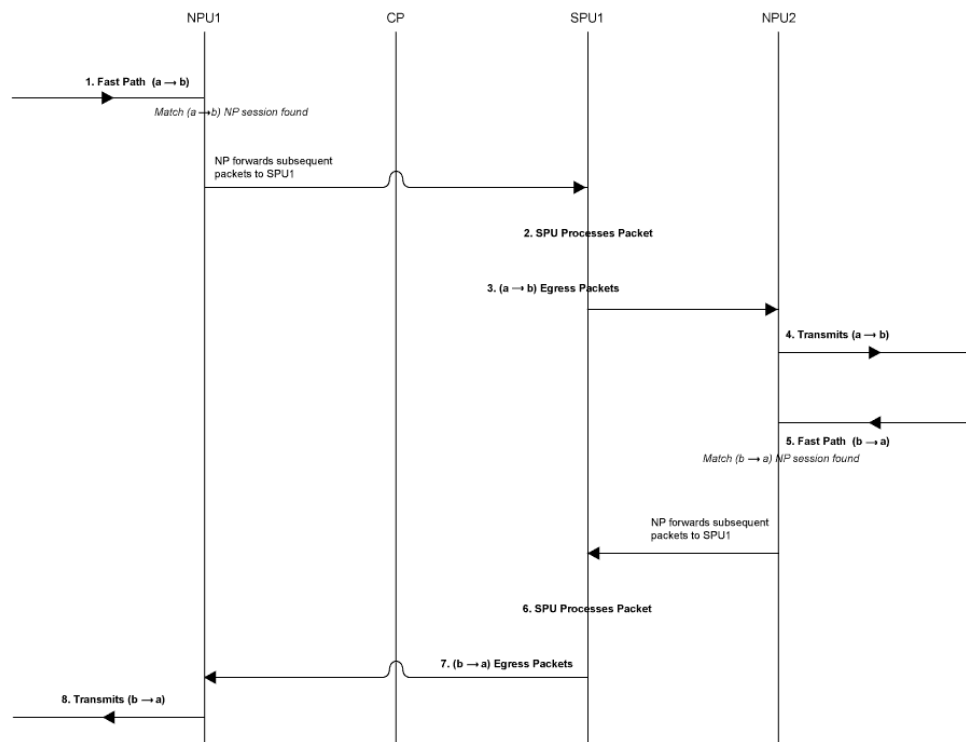
***8. The Interface Transmits the Packet from the Device.***

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 5 on page 20](#) illustrates the process a packet undergoes when it reaches the device and a session exists for the flow that the packet belongs to.

Figure 5: 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:  
$$\text{Global\_port\_index} = (\text{global\_pic} * 16) + \text{port\_offset}$$
- 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.

## Understanding Central Point Session Limit Performance Enhancements

---

**Supported Platforms**    **SRX Series, vSRX**

The central point architecture, which is supported on the SRX5000 Series devices, prevents data packets from going through the central point by offloading traffic management to Services Processing Units (SPUs). The system session capacity is extended because the session on the central point is removed. The session limit handles packet processing on the SPU and the central point, and updates the session limit counter when the session ages out.

Every session that ages out triggers a central point delete message; many such delete messages consume even more time. Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, the session limit performance is enhanced by replacing the central point delete message with a specific screen report message, thereby saving the central point memory capacity. The central point delete message contains many fields that are not useful for the session limit. So, this message is replaced with the specific screen report message, which only contains the necessary fields.



**NOTE:** Generating screen report message will traverse all the session limit entry and uses the changed session limit entry to generate the screen report message. For example, if 100 sessions from the same IP address age out, then all sessions have the same IP address and all sessions match the same session limit entry. Therefore this entry can be used to generate one report message instead of 100 delete messages. This method also helps reduce the number of exchange messages.

#### Release History Table

Release	Description
15.1X49-D40	Starting in Junos OS Release 15.1X49-D40 and Junos OS Release 17.3R1, the session limit performance is enhanced by replacing the central point delete message with a specific screen report message, thereby saving the central point memory capacity.

#### Related Documentation

- [Understanding Enhancements to Central Point Architecture for the SRX5000 Line on page 9](#)

## SRX3000 Line and SRX1400 Devices Processing Overview

**Supported Platforms** [SRX Series, vSRX](#)

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 on page 23](#)
- [Understanding the Data Path for Unicast Sessions on page 24](#)
- [Session Lookup and Packet Match Criteria on page 24](#)
- [Understanding Session Creation: First Packet Processing on page 24](#)
- [Understanding Fast-Path Processing on page 26](#)

## 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. For more information, see [“Understanding SRX Series Services Gateways Central Point Architecture” on page 7](#).

- 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`.

### Related Documentation

- [Juniper Networks Devices Processing Overview on page 3](#)
- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)

---

## SRX210 and SRX320 Services Gateway Processing Overview

### Supported Platforms [SRX Series](#)

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 on page 27](#)
- [Understanding First-Packet Processing on page 27](#)
- [Understanding Session Creation on page 27](#)
- [Understanding Fast-Path Processing on page 28](#)

## 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.

- Related Documentation**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)

---

## Understanding Central Point Architecture Flow Support for GTP and SCTP

**Supported Platforms**    [SRX Series, vSRX](#)

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, SRX 5600, and SRX 5800 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.

To prevent 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; otherwise, the session is migrated to the designated SPU.

To handle load-balancing issues, tag-based hash distribution is used to ensure even distribution of SCTP traffic from different associations 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 connection ID remains 0 if the connection tag is not used by the sessions.

#### Release History Table

Release	Description
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).

#### Related Documentation

- [Understanding Enhancements to Central Point Architecture for the SRX5000 Line on page 9](#)
- *enable-gtpu-distribution*

## Understanding the Flow Session Connection Filter Option

#### Supported Platforms [SRX Series, vSRX](#)

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 GPRS 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.

**Related  
Documentation**

- [Understanding Central Point Architecture Flow Support for GTP and SCTP on page 28](#)
- [Understanding Enhancements to Central Point Architecture for the SRX5000 Line on page 9](#)

## PART 2

# Configuring Flow-Based Sessions

- [Configuring Security Flow Sessions on page 33](#)





## CHAPTER 2

# Configuring Security Flow Sessions

- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)
- [Understanding Aggressive Session Aging on page 34](#)
- [Example: Controlling Session Termination for SRX Series Services Gateways on page 35](#)
- [Understanding TCP Session Checks per Policy on page 37](#)
- [Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways on page 38](#)
- [Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways on page 39](#)
- [Example: Configuring TCP Packet Security Checks Per Policy on page 41](#)
- [TCP Out-of-State Packet Drop Logging Overview on page 42](#)
- [Supported TCP Out-of-State Logging Features on page 44](#)
- [Example: Configuring TCP Out-of-State Packet Logging on SRX Series Devices on page 45](#)
- [Configuring the Timeout Value for Multicast Flow Sessions on page 48](#)
- [Clearing Sessions for SRX Series Services Gateways on page 50](#)
- [Understanding the Default Processing Behavior for IPv4 Traffic on page 50](#)
- [Understanding How Preserving Incoming Fragmentation Characteristics Can Improve Throughput on page 52](#)
- [Understanding ECMP Flow-Based Forwarding on page 53](#)
- [Understanding ECMP Flow-Based Forwarding for Reverse Traffic on SRX Series Devices and vSRX Instances on page 55](#)
- [Example: Configuring ECMP Flow-Based Forwarding on page 57](#)

## Understanding Session Characteristics for SRX Series Services Gateways

---

### Supported Platforms [SRX Series, vSRX](#)

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 to from 2 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(nowledge) 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

#### Related Documentation

- [Juniper Networks Devices Processing Overview on page 3](#)
- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
- [Clearing Sessions for SRX Series Services Gateways on page 50](#)
- [Example: Controlling Session Termination for SRX Series Services Gateways on page 35](#)

---

## Understanding Aggressive Session Aging

### Supported Platforms [SRX Series, vSRX](#)

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.)

#### Related Documentation

- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)
- [early-ageout on page 230](#)
- [high-watermark on page 246](#)
- [low-watermark on page 256](#)

## Example: Controlling Session Termination for SRX Series Services Gateways

**Supported Platforms** [SRX Series, vSRX](#)

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 on page 35](#)
- [Overview on page 35](#)
- [Configuration on page 36](#)
- [Verification on page 36](#)

### Requirements

Before you begin, understand the circumstances for terminating sessions. See [“Understanding Session Characteristics for SRX Series Services Gateways” on page 33](#).

### 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-timeout 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.

- Related Documentation**
- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)
  - [Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways on page 38](#)
  - [Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways on page 39](#)

## Understanding TCP Session Checks per Policy

**Supported Platforms** [SRX Series, vSRX](#)

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.

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. This command is also supported in Express Path mode.

### Related Documentation

- [Example: Configuring TCP Packet Security Checks Per Policy on page 41](#)

---

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

**Supported Platforms** [SRX Series, vSRX](#)

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

- [Requirements on page 38](#)
- [Overview on page 38](#)
- [Configuration on page 39](#)
- [Verification on page 39](#)

## Requirements

Before you begin, understand the circumstances for disabling TCP packet security checks. See “[Understanding Session Characteristics for SRX Series Services Gateways](#)” on [page 33](#).

## 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.

**Related Documentation**

- [Example: Controlling Session Termination for SRX Series Services Gateways on page 35](#)
- [Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways on page 39](#)

---

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

---

**Supported Platforms** [SRX Series, vSRX](#)

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

- [Requirements on page 40](#)
- [Overview on page 40](#)
- [Configuration on page 40](#)
- [Verification on page 41](#)

## Requirements

Before you begin, understand the circumstances for setting the maximum segment size. See [“Understanding Session Characteristics for SRX Series Services Gateways” on page 33.](#)

## 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;
  }
}
```

### Related Documentation

- [Example: Controlling Session Termination for SRX Series Services Gateways on page 35](#)
- [Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways on page 38](#)

## Example: Configuring TCP Packet Security Checks Per Policy

**Supported Platforms** [SRX Series, vSRX](#)

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

- [Requirements on page 41](#)
- [Overview on page 42](#)
- [Configuration on page 42](#)
- [Verification on page 42](#)

## Requirements

Before you begin, you must disable the tcp options, **tcp-syn-check**, and **tcp-sequence-check** that are configured at global level. See [“Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways” on page 38](#).

## 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.

### Related Documentation

- [Understanding TCP Session Checks per Policy on page 37](#)
- [Example: Disabling TCP Packet Security Checks for SRX Series Services Gateways on page 38](#)
- [Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways on page 39](#)

---

## TCP Out-of-State Packet Drop Logging Overview

---

### Supported Platforms [SRX Series](#)

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

### Supported Platforms [SRX Series](#)

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 on page 44](#)
- [Throttle Component on page 44](#)
- [Flexibility for Changing the Log Generation Rate on page 45](#)

### 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.

---

### Example: Configuring TCP Out-of-State Packet Logging on SRX Series Devices

---

#### Supported Platforms [SRX Series](#)

This example shows how to configure TCP out-of-state packet logging on SRX Series devices.

- [Requirements on page 45](#)
- [Overview on page 45](#)
- [Configuration on page 46](#)

#### Requirements

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

#### Overview

In this example, you enable TCP out-of-state packet logging and configure the packet log filter and the throttle component for controlling the logging rate. In addition, this

configuration provides you with the details for enabling TCP out-of-state logging on logical systems.

## Configuration

To configure this example, perform the following tasks:

**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 packet-log enable
set security flow packet-log packet-filter f1 source-prefix 1.1.1/32 destination-prefix
  2.2.2/32 source-port 10000 destination-port 2000 protocol tcp
set security flow packet-log throttle-interval 32
set security log mode stream
set logical-system LSYS1 security flow packet-log enable
set logical-system LSYS1 security flow packet-log throttle-interval 128
set security flow packet-log packet-filter f2 source-prefix 8.8.8.2/32 destination-prefix
  9.9.9.2/32 source-port 20000 destination-port 8000 protocol tcp
```

**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*.

To configure TCP out-of-state packet drop logging:

1. Enable packet logging in flow module.

```
[edit]
user@host# set security flow packet-log enable
```

2. Configure packet filters to generate logs for the target traffic.

```
[edit]
user@host# set security flow packet-log packet-filter f1 source-prefix 1.1.1/32
  destination-prefix 2.2.2/32 source-port 10000 destination-port 2000 protocol
  tcp
```

3. Configure a throttle interval to control the logging rate.

```
[edit]
user@host# set security flow packet-log throttle-interval 32
```



**NOTE:** The throttle interval is in milliseconds and should be configured as a power of two (0..32768).

---

4. Enable TCP out-of-state logging on the device to save the logs on the server.

```
[edit]
```

```
user@host# set security log mode stream
```

5. Configure TCP out-of-state logging on user logical systems.



**NOTE:** TCP out-of-state logging can be enabled per logical system each with a with different throttle interval. All the user logical systems share the same global packet-filter configuration with root logical system.

```
[edit]
user@host# set logical-system LSYS1 security flow packet-log enable
user@host# set logical-system LSYS1 security flow packet-log throttle-interval 128
user@host# set security flow packet-log packet-filter f2 source-prefix 8.8.8.2/32
    destination-prefix 9.9.9.2/32 source-port 20000 destination-port 8000 protocol
    tcpset
user@host# set logical-system LSYS1
user@host# set security log mode stream
```

**Results** From configuration mode, confirm the configuration of the interface by entering the **show security** and **show security log | display set** configuration mode command. If the command output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@host# show security
log {
  mode stream;
}
flow {
  packet-log {
    enable;
    throttle-interval 32;
    packet-filter f1 {
      protocol tcp;
      source-prefix 1.1.1.1/32;
      destination-prefix 2.2.2.2/32;
      source-port 10000;
      destination-port 2000;
    }
  }
}
zones {
  security-zone Host {
    host-inbound-traffic {
      system-services {
        all;
      }
    }
  }
  interfaces {
    ge-0/0/0.0;
  }
}
```

```
}  
}  
[edit]  
user@host# show security log | display set  
set security log mode stream
```

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

---

## Verification

### *Verify the Configuration*

**Purpose** Verify that the configuration is correct.

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

---

## Configuring the Timeout Value for Multicast Flow Sessions

**Supported Platforms** [SRX Series](#), [vSRX](#)

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 multicast 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**
- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)
  - [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)

---

## Clearing Sessions for SRX Series Services Gateways

**Supported Platforms** [SRX Series, vSRX](#)

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 on page 50](#)
- [Terminating a Specific Session for SRX Series Services Gateways on page 50](#)
- [Using Filters to Specify the Sessions to Be Terminated for SRX Series Services Gateways on page 50](#)

### 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
```

- See Also**
- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)

---

## Understanding the Default Processing Behavior for IPv4 Traffic

**Supported Platforms** [SRX Series, vSRX](#)

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.

Release History Table

Release	Description
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 <i>not</i> 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 <i>must</i> reboot the device when switching between flow mode, packet mode, and drop mode.

Related Documentation

- [Juniper Networks Devices Processing Overview on page 3](#)
- [Flow-Based Processing for IPv6 Traffic on page 132](#)

## Understanding How Preserving Incoming Fragmentation Characteristics Can Improve Throughput

---

### Supported Platforms [SRX Series, vSRX](#)

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 sets the egress interface MTU size to the smallest value. 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 with the following statement to enable the feature that takes into account the size of incoming packet fragments:

```
user@host# set security flow preserve-incoming-frag-size
```

Table 5 on page 53 summarizes how the SRX Series egress MTU size is determined.

**Table 5: 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 <i>smaller</i> than the existing egress MTU size.	And the egress MTU size is <i>larger</i> than the maximum incoming fragment size.	Incoming fragment size is used.
If the largest fragment is <i>larger</i> than the existing egress MTU size.	And the egress MTU size is <i>smaller</i> than the maximum incoming fragment size	Existing MTU size is used.



**NOTE:** This feature is supported on SRX Series devices. It supports through-traffic and traffic exiting a tunnel. It 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. Here is a summary of how path MTU discovery works:



**NOTE:** For stream-based applications—which are application layer gateway (ALG) applications—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.

Related  
Documentation

- [preserve-incoming-fragment-size on page 268](#)

## Understanding ECMP Flow-Based Forwarding

Supported Platforms [SRX Series](#)

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; 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.

**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.

**Related Documentation**

- [Example: Configuring ECMP Flow-Based Forwarding on page 58](#)
- [Understanding Routing Policies](#)
- [Categories of Routing Policy Match Conditions](#)
- [Summary of Routing Policy Actions](#)

## Understanding ECMP Flow-Based Forwarding for Reverse Traffic on SRX Series Devices and vSRX Instances

**Supported Platforms** [SRX Series](#)

This topic provides a brief overview of equal-cost multipath (ECMP) for 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, see [“Understanding ECMP Flow-Based Forwarding” on page 53](#).

- [ECMP Overview on page 55](#)
- [ECMP Implementation for Junos OS SRX Series Devices and vSRX Instances on page 55](#)
- [ECMP for Reverse Traffic on page 56](#)

**ECMP Overview**

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.

**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.

Release History Table

Release	Description
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.

## Example: Configuring ECMP Flow-Based Forwarding

**Supported Platforms** [M Series](#), [MX Series](#), [QFabric System](#), [QFX Series](#), [SRX Series](#), [T Series](#)

- [Understanding ECMP Flow-Based Forwarding on page 57](#)
- [Example: Configuring ECMP Flow-Based Forwarding on page 58](#)

## Understanding ECMP Flow-Based Forwarding

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; 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.

- See Also**
- [Example: Configuring ECMP Flow-Based Forwarding on page 58](#)
  - [Understanding Routing Policies](#)
  - [Categories of Routing Policy Match Conditions](#)
  - [Summary of Routing Policy Actions](#)

## Example: Configuring ECMP Flow-Based Forwarding

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

- [Requirements on page 58](#)
- [Overview on page 59](#)
- [Configuration on page 59](#)
- [Verification on page 63](#)

### Requirements

---

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

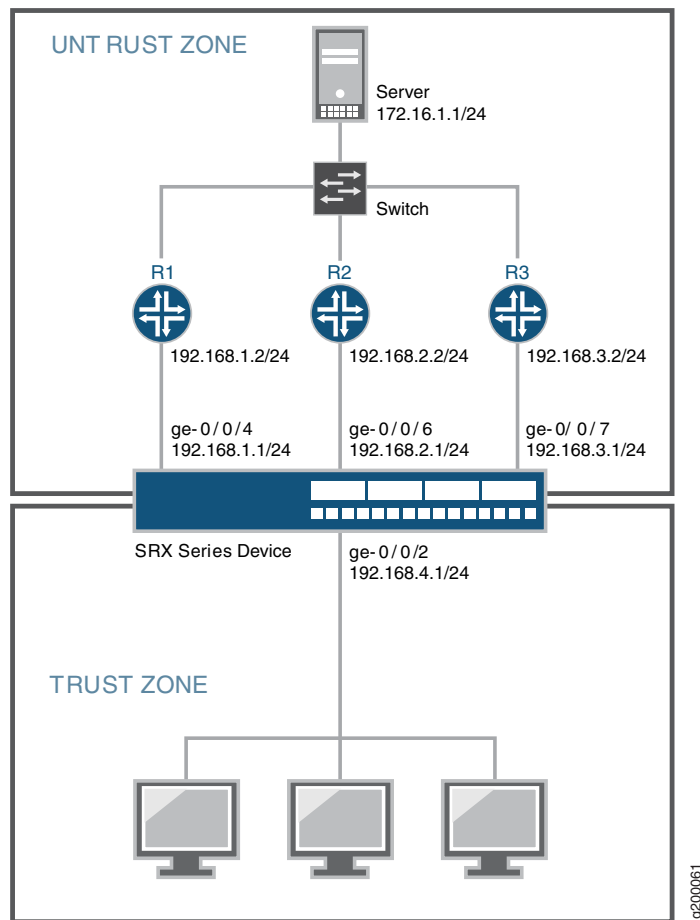
## Overview

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 6 on page 59 shows the topology used in this example.

Figure 6: 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. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

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          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.

**See Also** • [Understanding ECMP Flow-Based Forwarding on page 53](#)

### 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.

**Related Documentation**

- *Configuring ECMP Next Hops for RSVP and LDP LSPs for Load Balancing*
- *Configuring ECMP-Aware BFD for LDP LSPs*





## PART 3

# Improving Flow-Based Performance

- [Expanding Session Capacity by Device on page 67](#)
- [Managing Sessions and Flow Distribution on page 71](#)
- [Reducing Long Packet-Processing Latency by Express Path on page 79](#)
- [Managing IPsec Tunnel Fragmentation on page 121](#)



# Expanding Session Capacity by Device

- [Expanding Session Capacity by Device on page 67](#)
- [Expanding Session Capacity on an SRX3400 or SRX3600 Device on page 68](#)
- [Reverting to Default Session Capacity on an SRX5800 Device on page 68](#)
- [Verifying the Current Session Capacity on page 68](#)

## Expanding Session Capacity by Device

**Supported Platforms** [SRX Series, vSRX](#)

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 6 on page 67](#) 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 6: Maximum Central Point Session Increases**

SRX Series Devices	Maximum Concurrent Sessions on a Fully Loaded System	
	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

**Related Documentation**

- [Reverting to Default Session Capacity on an SRX5800 Device on page 68](#)

---

## Expanding Session Capacity on an SRX3400 or SRX3600 Device

**Supported Platforms** [SRX Series](#)

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. For license installation details, see *Administration Guide for Security Devices*.
2. Reboot the device to implement the expanded session capacity.

**Related Documentation**

- [Expanding Session Capacity by Device on page 67](#)

---

## Reverting to Default Session Capacity on an SRX5800 Device

**Supported Platforms** [SRX Series](#), [vSRX](#)

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. Reboot the device to implement the new value.

---

## Verifying the Current Session Capacity

**Supported Platforms** [SRX Series](#), [vSRX](#)

**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.



## CHAPTER 4

# Managing Sessions and Flow Distribution

- [Understanding Load Distribution in SRX5800, SRX5600, and SRX5400 Devices on page 71](#)
- [Understanding Packet-Ordering Function on SRX5000 Line Devices on page 73](#)
- [Disabling Packet-Ordering Mode on SRX5000 Line Devices on page 74](#)
- [Understanding Session Distribution on High End SRX Series Devices in Adaptive Mode on page 75](#)

## Understanding Load Distribution in SRX5800, SRX5600, and SRX5400 Devices

---

### Supported Platforms [SRX Series, vSRX](#)

The load distribution algorithm, which is supported on the SRX5800, SRX 5600, and SRX 5400 devices and vSRX, 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 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.

#### Revision History

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.

### 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.

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

#### Release History Table

Release	Description
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.

## Understanding Packet-Ordering Function on SRX5000 Line Devices

### Supported Platforms [SRX Series, vSRX](#)

The packet-ordering function, which is supported on the SRX 5400, SRX 5600, and SRX 5800, 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. See [“Disabling Packet-Ordering Mode on SRX5000 Line Devices” on page 74](#) for more details.

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.

The packet-ordering functionality using the Packet Ordering Engine is supported on SRX5400, SRX5800 and SRX5600 devices with next-generation SPCs.

**Related  
Documentation**

- [Understanding SRX Series Services Gateways Central Point Architecture on page 7](#)
- [Disabling Packet-Ordering Mode on SRX5000 Line Devices on page 74](#)
- [packet-ordering-mode \(Application Services\) on page 266](#)

---

## Disabling Packet-Ordering Mode on SRX5000 Line Devices

**Supported Platforms**   [SRX Series, vSRX](#)

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)
```

**Related Documentation** • [Understanding Packet-Ordering Function on SRX5000 Line Devices on page 73](#)

## [Understanding Session Distribution on High End SRX Series Devices in Adaptive Mode](#)

**Supported Platforms** [SRX Series, vSRX](#)

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:

$$W_i = \text{Sum}(W_{1-n}) * C_i * S_i / \text{Sum}(C_{1-n} * S_{1-n})$$

Where:

- **W<sub>i</sub>**— weight assigned to the SPU.
- **Sum(W<sub>1-n</sub>)**— Total weight of system. This values is constant.
- **n**—total number of SPUs.
- **C<sub>i</sub>**—available CPU computational power of the SPU.
- **S<sub>i</sub>**—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
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.



## CHAPTER 5

# Reducing Long Packet-Processing Latency by Express Path

- [Understanding Session Cache on page 79](#)
- [Express Path Overview on page 83](#)
- [Understanding the Services Offload Solution on page 99](#)
- [Enabling and Disabling Express Path on page 100](#)
- [Example: Enabling Express Path in Security Policies on page 102](#)
- [Example: Configuring an NPC on SRX3000 Line Devices or SRX1400 Devices to Support Express Path on page 103](#)
- [Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path on page 105](#)
- [Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path on page 107](#)
- [Example: Configuring SRX5K-MPC3-100G10G \(IOC3\) and SRX5K-MPC3-40G10G \(IOC3\) on an SRX5000 Line Device to Support Express Path on page 109](#)
- [Example: Configuring Express Path on an SRX5000 Line Device with IOC3 for IPv6 Traffic on page 112](#)
- [Example: Configuring Low Latency on page 117](#)

## Understanding Session Cache

---

### Supported Platforms [SRX Series, vSRX](#)

- [Overview on page 79](#)
- [Selective Session Cache Installation on page 81](#)
- [IPsec VPN Session Affinity Enhancement Using Session Cache on page 82](#)
- [Fragmentation Packet Ordering Using NP Session Cache on page 82](#)

### 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.

The following three priority levels are used to determine which type of traffic can install session cache on the IOCs:

- **Priority 1 (P1)**—Express Path qualified traffic
- **Priority 2 (P2)**—IPsec tunneling, Fragmentation ordering, and NAT/SZ traffic
- **Priority 3 (P3)**—All other types of traffic

The IOCs maintain and monitor session cache usage threshold levels. Session cache usage less than 25 percent is defined as “green,” 26 to 50 percent is “yellow,” 51 to 99 percent is “orange,” and 99 percent and above is defined as “red.” The IOCs update current real-time session cache usage to the SPU. The SPU requests the IOC to install the session cache for selective high-priority traffic sessions. When the session cache usage is green, the session cache will be installed for all types of traffic. When the usage is yellow, session can be installed only for certain sessions including IPsec, Fragmentation, and NAT/SZ traffic sessions, and when the usage is orange, only Express Path qualified sessions will be installed. When the session cache usage is red, NP cache and the Express Path sessions are not allowed to install from SPU to the IOCs.

**Table 7: Session Cache Installation Bars**

Traffic Type	Green	Yellow	Orange	Red
Express Path traffic	Yes	Yes	Yes	No
IPsec, Fragmentation, and 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 (see [Table 8 on page 82](#)) 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.

Table 8 on page 82 shows the cache table utilization (CTU) bits and the respective session cache table utilization.

**Table 8: Session Cache Table Utilization Bits Status**

Session Cache Table Utilization (CTU) Bits	IOC Session Cache/Express Path Table Utilization	Action
00	0% < utilization < 25%	Flowd installs any eligible session.
01	25% < utilization < 75%	Flowd installs only high-priority sessions, such as Express Path, IPsec, IPsec clear-text, NAT, and fragmented sessions.
10	75% < utilization < 100%	Flowd installs only Express Path sessions.
11	NP cache/Express Path table is full	Flowd stops installing sessions.

## 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.

#### Release History Table

Release	Description
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 Overview on page 83](#)
- [Understanding VPN Session Affinity](#)

## Express Path Overview

#### Supported Platforms

SRX Series, vSRX

- [Understanding Express Path Functionality on page 84](#)
- [Understanding Express Path Support on SRX Series Devices on page 85](#)
- [Understanding Express Path Features on page 86](#)
- [Express Path Limitations on page 91](#)
- [Express Path Support on NP-IOC Card on page 92](#)
- [Express Path Support on SRX5K Modular Port Concentrator on page 93](#)
- [Express Path Support on SRX5K-MPC3-100G10G \(IOC3\) and SRX5K-MPC3-40G10G \(IOC3\) on page 94](#)
- [IPv6 Flow in Express Path Mode for IOC2 and IOC3 on page 96](#)
- [IPv6 Flow in Express Path Mode on page 97](#)

## Understanding Express Path Functionality

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 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 9 on page 85 provides details about the Express Path support on different SRX Series cards.

**Table 9: Express Path Support on SRX Series Device Cards**

SRX Series Device	Card Name and Model Number	Earliest Supported Release
<b>SRX5000 Line Devices I/O Cards (IOCs)</b>		
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: <ul style="list-style-type: none"> <li>SRX-IOC-16GE-TX</li> <li>SRX-IOC-4XGE-XFP</li> <li>SRX-IOC-16GE-SFP</li> </ul>	Junos OS Release 11.4
SRX5400, SRX5600, SRX5800	SRX5K-MPC containing one of the following MICs: <ul style="list-style-type: none"> <li>SRX-MIC-10XGE-SFFP</li> <li>SRX-MIC-2X40GE-OSFP</li> <li>SRX-MIC-1X100GE-CFP</li> <li>SRX-MIC-20GE-SFP</li> </ul>	Junos OS Release 12.3X48-D10
SRX5400, SRX5600, SRX5800	SRX5K-MPC3 (IOC3) containing one of the following MPCs: <ul style="list-style-type: none"> <li>SRX5K-MPC3-40G10G (24x10GE + 6x40GE MPC)</li> <li>SRX5K-MPC3-100G10G (2x100GE + 4x10GE MPC)</li> </ul>	Junos OS Release 15.1X49-D10
<b>SRX1400 and SRX3000 Line Devices I/O Cards (IOCs)</b>		
SRX1400, SRX3400, SRX3600	SRX3K-16GE-TX	Junos OS Release 11.4
SRX1400, SRX3400, SRX3600	SRX3K-16GE-SFP	Junos OS Release 11.4

Table 9: Express Path Support on SRX Series Device Cards (*continued*)

SRX Series Device	Card Name and Model Number	Earliest Supported Release
SRX1400, SRX3400, SRX3600	SRX3K-2XGE-XFP	Junos OS Release 11.4
<b>SRX1400 and SRX3000 Line Devices NP-IOC</b>		
SRX1400, SRX3400, SRX3600	SRX1K3K-NP-2XGE-SFPP	Junos OS Release 12.1X44-D10
<b>SRX3000 Line Devices Switch Fabric Board</b>		
SRX3400, SRX3600	SRX3K-SFB-12GE	Junos OS Release 11.4
<b>SRX1400 Devices System I/O Cards (SYSIOCs)</b>		
SRX1400	SRX1K-SYSIO-GE	Junos OS Release 11.4
SRX1400	SRX1K-SYSIO-XGE	Junos OS Release 11.4



**NOTE:** Different Express Path features are supported on different cards for different Junos OS releases. See the *Junos OS Release Notes* for details.



**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 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 10 on page 87](#) displays the total number of session wings, including both Express Path and non-Express Path.

**Table 10: 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
FIOC	2.3 million	198,000	900,000	600,000	400,000

**Table 10: 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	
SRX3000 line device/NP-IOC	2.3 million	2.3 million	2.3 million	2 million	1.5 million
SRX3000 line device/NPC	2.3 million	512,000	256,000	170,000	128,000
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

### 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.

### 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.
  - 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. Currently, in the SRX3000 line all interfaces are assigned an internal global port number for interface configuration and management purposes. However, this method limits the slot positions in which an IOC card can be inserted (that is, an IOC can only be present in slot numbers 0–3). An NP-IOC can be present in any of the slots, meaning that there also can be an interface at any slot.



**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 loading*), 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 loading*) 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.

Express Path support on IOC3 is illustrated in [Figure 7 on page 95](#), [Figure 8 on page 96](#), and [Figure 9 on page 96](#).

**Figure 7: IOC3 Intra-PFE Express Path**

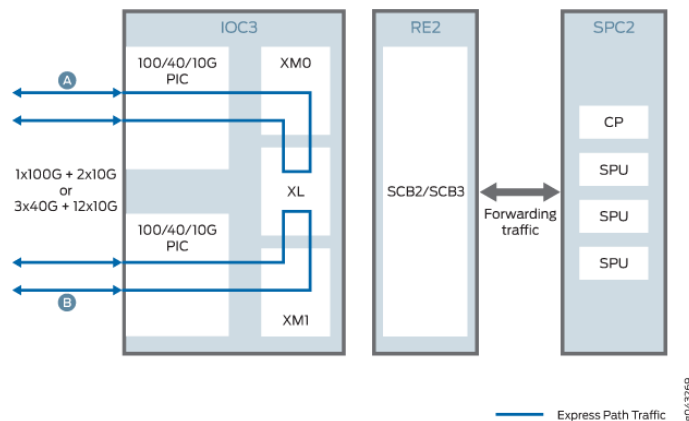


Figure 8: IOC3 Inter-PFE Express Path

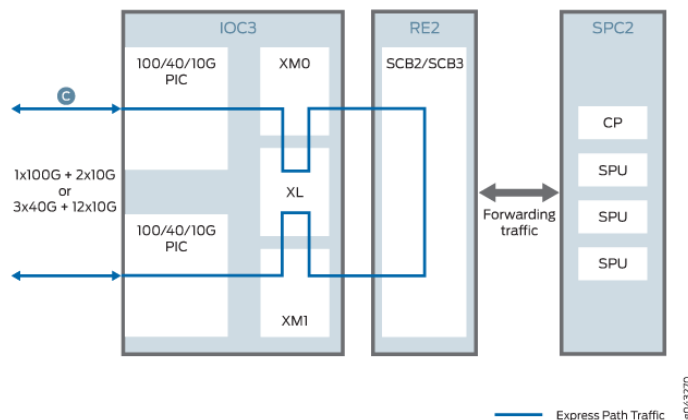
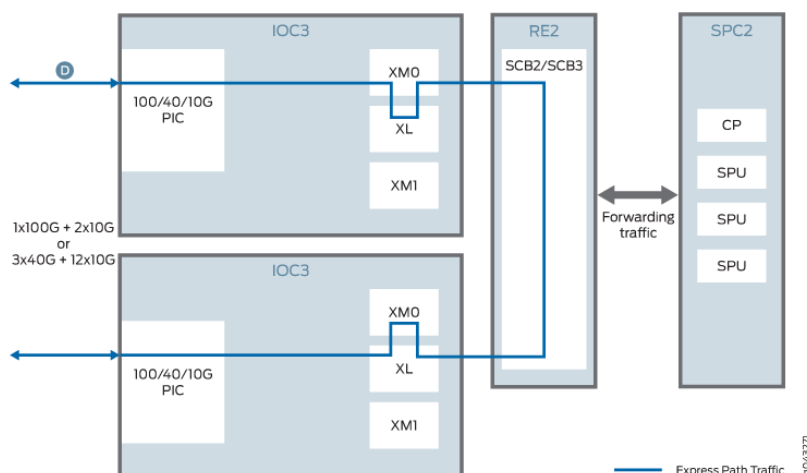


Figure 9: Inter-IOC3 Express Path



## 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 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.



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.
- Express Path is not supported in IPsec VPN, and IDP configurations. Normal flow sessions will be used in these scenarios.
- Express Path is not supported in logical system configurations, because logical systems use the sessions installed from the normal flow sessions.
- Multicast sessions with multiple fan-outs are not supported in Express Path mode. A normal session is installed if a multicast session with more than one fan-out exists.
- Active/active chassis cluster configuration is not supported in Express Path mode. Only active/passive chassis cluster configuration is supported.
- 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.
- Fragmented packets are not supported in Express path mode. IPv6 fragmented packets will be directed to the SPU and hence they will not be qualified for Express Path.

See [“Example: Configuring Express Path on an SRX5000 Line Device with IOC3 for IPv6 Traffic” on page 112](#) for details on configuring Express Path for IPv6 traffic.

## 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 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.

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 are 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.

**Release History Table**

Release	Description
15.1X49-D80	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.
15.1X49-D60	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.
15.1X49-D40	Starting in Junos OS 15.1X49-D40 and Junos OS Release 17.3R1, IPv6 is supported.
12.3X48-D10	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.

**Related Documentation**

- [Example: Configuring Low Latency on page 117](#)
- [Understanding the Services Offload Solution on page 99](#)
- [Enabling and Disabling Express Path on page 100](#)

## Understanding the Services Offload 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 address 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 than 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. The network processor does not change session status even after the FIN or RST packet is received. It continues to receive and process any packets during the transit session.

**Related  
Documentation**

- *Junos OS Feature Support Reference for SRX Series and J Series Devices*
- [Enabling and Disabling Express Path on page 100](#)

---

## Enabling and Disabling Express Path

---

**Supported Platforms**    [SRX Series, vSRX](#)



**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.”)

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.



**NOTE:** For SRX5600 and SRX5800 devices, enable Express Path on the IOC PICor SRX5K modular port concentrators (SRX5K-MPC), SRX5K-MPC3-100G10G (IOC3), and SRX5K-MPC3-40G10G (IOC3) on SRX5000 line devices, enable Express Path on the FPCs.



**NOTE:** When device is operating in chassis cluster mode, you need to reboot both the nodes when changing FPC(s) to Express Path mode.

To configure the Express Path mode:

- For SRX3400 and SRX3600 devices, use the **set chassis fpc fpc-number pic pic-number services-offload** command.



**NOTE:** 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.

- For configuring Express Path on an SRX5000 line device, 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. Then configure the security policy to determine if the session is for Express Path.



**NOTE:** The **set chassis fpc *fpc-number* services-offload** command is deprecated.

- To disable Express Path on SRX3000 line and SRX5000 line devices, use the **delete chassis fpc *fpc-number* pic *pic-number* services-offload** command.
- To disable Express Path on an SRX5000 line device with Modular Port Concentrator (MPC), use the **delete chassis fpc *fpc-number* np-cache** command.



**NOTE:** The **delete chassis fpc *fpc-number* services-offload** command is deprecated.



**NOTE:** You need to reboot the device when you disable Express Path.

System log files are stored locally on the device in the default `/var/log/security` directory. You can use system log files to retrieve information about the Express Path configuration changes.

Release History Table

Release	Description
12.3X48-D10	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.

#### Related Documentation

- [Understanding the Services Offload Solution on page 99](#)
- [Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path on page 105](#)
- [Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path on page 107](#)

## Example: Enabling Express Path in Security Policies

---

**Supported Platforms** [SRX Series, vSRX](#)

This example shows how to enable Express Path (formerly known as *services offloading*) in security policies.

- [Requirements on page 102](#)
- [Overview on page 102](#)
- [Configuration on page 102](#)
- [Verification on page 103](#)

### Requirements

Before you begin, see “[Express Path Overview](#)” on [page 83](#).

### 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.

**Related Documentation**

- [Express Path Overview on page 83](#)

## Example: Configuring an NPC on SRX3000 Line Devices or SRX1400 Devices to Support Express Path

---

**Supported Platforms** [SRX Series, vSRX](#)

This example shows how to configure an NPC on the SRX3000 line of devices to support Express Path (formerly known as *services offloading*).

- [Requirements on page 104](#)
- [Overview on page 104](#)
- [Configuration on page 104](#)
- [Verification on page 105](#)

## Requirements

Before you begin, see “Express Path Overview” on page 83.

## Overview

In this example, you configure an NPC on SRX3000 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 6 pic 0 services-offload
```

**Step-by-Step Procedure** To configure the NPC on an SRX3000 line device for Express Path:

1. Enable the Express Path mode on the NPC.

```
[edit]
user@host# set chassis fpc 6 pic 0 services-offload
```



**NOTE:** For SRX3000 line devices, the NPC slot number is 6.

2. Commit the configuration.

```
[edit]
user@host# commit
warning: System restart is required after fpc 6 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      SRX3k SFB 12GE
  PIC 0  Online      8x 1GE-TX 4x 1GE-SFP
Slot 1  Online      SRX3k 16xGE SFP
  PIC 0  Online      16x 1GE-SFP
Slot 5  Online      SRX3k SPC
  PIC 0  Online      SPU Cp-Flow
```



```
Slot 6  Online      SRX3k NPC
PIC 0   Online      NPC PIC- services-offload
```

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

## Verification

### Verifying the Configuration of NPC for Express Path

**Purpose** Verify that the NPC was configured properly for Express Path.

**Action** From operational mode, enter the **show chassis fpc pic-status** command.

**Related Documentation**

- [Express Path Overview on page 83](#)
- [Example: Enabling Express Path in Security Policies on page 102](#)

## Example: Configuring an IOC on SRX5000 Line Devices to Support Express Path

**Supported Platforms** [SRX Series, vSRX](#)

This example shows how to configure an IOC on SRX5000 line of devices to support Express Path (formerly known as *services offloading*).

- [Requirements on page 105](#)
- [Overview on page 105](#)
- [Configuration on page 105](#)
- [Verification on page 106](#)

## Requirements

Before you begin, see “Express Path Overview” on page 83.

## 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.

- Related Documentation**
- [Express Path Overview on page 83](#)
  - [Example: Enabling Express Path in Security Policies on page 102](#)

## Example: Configuring an SRX5K-MPC on an SRX5000 Line Device to Support Express Path

---

**Supported Platforms** [SRX Series, vSRX](#)

This example shows how to configure an SRX5K-MPC on an SRX5000 line device to support Express Path (formerly known as *services offloading*).

- [Requirements on page 107](#)
- [Overview on page 107](#)
- [Configuration on page 107](#)
- [Verification on page 108](#)

### 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, see “[Express Path Overview](#)” on page 83.

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.

- Related Documentation**
- [Express Path Overview on page 83](#)
  - [Example: Enabling Express Path in Security Policies on page 102](#)
  - [Enabling and Disabling Express Path on page 100](#)

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

**Supported Platforms** [SRX Series, vSRX](#)

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 on page 109](#)
- [Overview on page 110](#)
- [Configuration on page 110](#)
- [Verification on page 111](#)

### 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, see [“Express Path Overview”](#) on page 83.

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.

- Related Documentation**
- [Express Path Overview on page 83](#)
  - [Example: Enabling Express Path in Security Policies on page 102](#)
  - [Enabling and Disabling Express Path on page 100](#)
  - [Understanding Session Cache on page 79](#)

---

## Example: Configuring Express Path on an SRX5000 Line Device with IOC3 for IPv6 Traffic

---

**Supported Platforms** [SRX Series, vSRX](#)

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 for IPv6 traffic.

- [Requirements on page 112](#)
- [Overview on page 113](#)
- [Configuration on page 113](#)
- [Verification on page 115](#)

### 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, see “[Express Path Overview](#)” on [page 83](#).



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.

**Related Documentation**

- [Express Path Overview on page 83](#)
- [Example: Enabling Express Path in Security Policies on page 102](#)

- [Enabling and Disabling Express Path on page 100](#)
- [Understanding Session Cache on page 79](#)

## Example: Configuring Low Latency

### Supported Platforms [SRX Series, vSRX](#)

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 on page 117](#)
- [Overview on page 117](#)
- [Configuration on page 118](#)
- [Verification on page 118](#)

## Requirements

Before you begin, see “[Express Path Overview](#)” on [page 83](#).

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      SRX3k SFB 12GE
  PIC 0   Online      8x 1GE-TX 4x 1GE-SFP
Slot 1   Online      SRX3k 2x10GE XFP
  PIC 0   Online      2x 10GE-XFP
Slot 2   Online      SRX3k SPC
  PIC 0   Online      SPU Cp-Flow
Slot 7   Online      SRX1k3k 2x10GE NP-IOC
  PIC 0   Online      2x 10GE-SFP+- services-offload low-latency
```

- Related Documentation**
- [Express Path Overview on page 83](#)
  - [Example: Enabling Express Path in Security Policies on page 102](#)





## CHAPTER 6

# Managing IPsec Tunnel Fragmentation

- [Understanding the Fragmentation Counters Feature on page 121](#)

## Understanding the Fragmentation Counters Feature

---

**Supported Platforms** [SRX Series, vSRX](#)

Packet fragmentation degrades system performance and raises considerable concerns in regard to 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 on page 121](#)
- [Understanding Fragmentation and MTU and MSS Sizes on page 122](#)
- [Using Fragmentation Counter Statistics to Tune Your System on page 123](#)

## 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, see [“Example: Setting the Maximum Segment Size for All TCP Sessions for SRX Series Services Gateways” on page 39](#).

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 on page 555](#)
  - [show security flow statistics on page 564](#)

## PART 4

# Managing Flow-Based Processing for IPv6

- [Enabling IPv6 Flow-Based Processing on page 127](#)
- [Managing IPv6 Packets on page 141](#)
- [Configuring IPv6 Dual-Stack on page 151](#)



## CHAPTER 7

# Enabling IPv6 Flow-Based Processing

- [IPv6 Advanced Flow on page 127](#)
- [Understanding Sessions for IPv6 Flows on page 129](#)
- [Understanding IPv6 Flow Processing on SRX5400, SRX5600, and SRX5800 devices on page 129](#)
- [Flow-Based Processing for IPv6 Traffic on page 132](#)
- [Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways on page 134](#)

## IPv6 Advanced Flow

---

**Supported Platforms** [SRX Series, vSRX](#)

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.

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

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.

**Related  
Documentation**

- *Understanding IPv6 Address Space, Addressing, Address Format, and Address Types*
- [About IPv6 Packet Header Verification Performed by the Flow Module for SRX Series Devices on page 145](#)



## Understanding Sessions for IPv6 Flows

**Supported Platforms** [SRX Series, vSRX](#)

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).

**Related Documentation**

- [Understanding IPv6 Flow Processing on SRX5400, SRX5600, and SRX5800 devices on page 129](#)
- *Understanding IPv6 Address Space, Addressing, Address Format, and Address Types*

## Understanding IPv6 Flow Processing on SRX5400, SRX5600, and SRX5800 devices

**Supported Platforms** [SRX Series](#)

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 central point architecture it is hashed by the 3-tuple and the security parameter index (SPI) field, after enhancements to the central 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.

- Related Documentation**
- [Understanding Sessions for IPv6 Flows on page 129](#)
  - [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)

---

## Flow-Based Processing for IPv6 Traffic

---

### Supported Platforms [SRX Series, vSRX](#)

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, see *Interfaces Feature Guide for Security Devices*.

#### 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.

## 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 <i>not</i> 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 <i>must</i> reboot the device when switching between flow mode, packet mode, and drop mode.

## Related Documentation

- [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)
- [Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways on page 134](#)

## Using Filters to Display IPv6 Session and Flow Information for SRX Series Services Gateways

**Supported Platforms** SRX Series, vSRX

**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:dbf8:5::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:dbf8:5::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:dbf8:5::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:dbf8:5::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:dbf8:5::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:dbf8:5::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:dbf8:5::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-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**

- [Flow-Based Processing for IPv6 Traffic on page 132](#)
- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
- [Displaying a Summary of Sessions for SRX Series Services Gateways on page 165](#)
- [Displaying Session and Flow Information About Sessions for SRX Series Services Gateways on page 166](#)
- [Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways on page 166](#)
- [Information Provided in Session Log Entries for SRX Series Services Gateways on page 167](#)
- [Clearing Sessions for SRX Series Services Gateways on page 50](#)



## CHAPTER 8

# Managing IPv6 Packets

- [The IPv6 Packet Header and SRX Series Overview on page 141](#)
- [About the IPv6 Basic Packet Header on page 142](#)
- [Understanding IPv6 Packet Header Extensions on page 144](#)
- [About IPv6 Packet Header Verification Performed by the Flow Module for SRX Series Devices on page 145](#)
- [Understanding Path MTU Messages for IPv6 Packets on page 145](#)
- [Understanding How SRX Series Devices Handle Packet Fragmentation for IPv6 Flows on page 147](#)
- [Understanding How SRX Series Devices Handle ICMPv6 Packets on page 147](#)

## The IPv6 Packet Header and SRX Series Overview

---

### Supported Platforms [SRX Series, vSRX](#)

This topic identifies the IP version 6 (IPv6) packet header and its extensions and options.

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.

### Related Documentation

- [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)
- [About the IPv6 Basic Packet Header on page 142](#)

- [Understanding IPv6 Packet Header Extensions on page 144](#)

## About the IPv6 Basic Packet Header

---

**Supported Platforms** [SRX Series, vSRX](#)

This topic identifies the IP version 6 (IPv6) basic packet header fields with their bit lengths and uses.

Header Name	Bit Length	Purpose
Version	4	Specifies that IP version 6 is used. The IPv6 version field contains a value of 6 indicating that IPv6 is used, as opposed to 4 for IP version 4.
Traffic Class	8	Allows source nodes or routers to identify different classes (or priorities for quality of service) for IPv6 packets. (This field replaces the IPv4 Type of Service field.)
Flow Label	20	Identifies the flow to which the packet belongs. Packets in a flow share a common purpose, or belong to a common category, as interpreted by external devices such as routers or destination hosts.  <b>NOTE:</b> For IPv6 flow-based packets, Junos OS for SRX Series devices does not use the flow label field.
Payload Length	16	Specifies the length of the IPv6 packet payload, or contents, expressed in octets.

Header Name	Bit Length	Purpose
Next Header	8	<p>Identifies the type of Internet Protocol for the header that immediately follows the IPv6 header. The Next Header field replaces the IPv4 Protocol field. It is an optional field.</p> <p>This protocol can be one of two types:</p> <ul style="list-style-type: none"> <li>• An IPv6 extension header. For example, if the device performs IP security on exchanged packets, the Next Header value is probably 50 (ESP extension header) or 51 (AH extension header). Extension headers are optional.</li> <li>• An upper-layer Protocol Data Unit (PDU). For example, the Next Header value could be 6 (for TCP), 17 (for UDP), or 58 (for ICMPv6).</li> </ul> <p>The flow module processes these headers sequentially within the context of a packet flow.</p> <p>If it encounters one of the following extension headers, the software parses it and regards the packet as a corresponding protocol packet.</p> <ul style="list-style-type: none"> <li>• Internet Control Message Protocol version 6 (ICMPv6)</li> <li>• Transport Control Protocol (TCP)</li> </ul> <p><b>NOTE:</b> The device checks the TCP header length as part of its sanity checks.</p> <ul style="list-style-type: none"> <li>• UDP</li> </ul> <p><b>NOTE:</b> The device checks the UDP length as part of its sanity checks.</p> <ul style="list-style-type: none"> <li>• Enhanced Security Protocol (ESP) or Authentication Header (AH)</li> </ul>
Hop Limit	8	<p>Specifies the maximum number of hops the packet can make after transmission from the host device. When the Hop Limit value is zero, the device drops the packet and generates an error message. (This field is similar the to Time to Live IPv4 field.)</p>
Source IP Address	128	<p>Identifies the host device, or interface on a node, that generated the IPv6 packet.</p>
Destination IP Address	128	<p>Identifies the host device, or interface on a node, to which the IPv6 packet is to be sent.</p> <p><b>NOTE:</b> The destination address can appear twice, the first instance after the hop limit following the source IP address and the second instance after the final extension header.</p>

**Related  
Documentation**

- [Understanding IPv6 Packet Header Extensions on page 144](#)
- [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)

## Understanding IPv6 Packet Header Extensions

**Supported Platforms** SRX Series, vSRX

This topic defines IP version 6 (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 11 on page 144](#):



**NOTE:** The destination IP address can appear twice, once after the hop-by-hop header and again after the last extension header.

**Table 11: IPv6 Extension Headers**

Header Name	Purpose
Hop-by-Hop Options	Specifies delivery parameters at each hop on the path to the destination host.  <b>NOTE:</b> 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 11: 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.  <b>NOTE:</b> 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.

- Related Documentation**
- [About the IPv6 Basic Packet Header on page 142](#)
  - [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)

## About IPv6 Packet Header Verification Performed by the Flow Module for SRX Series Devices

**Supported Platforms** [SRX Series, vSRX](#)

This topic gives an overview of some of the IP version 6 (IPv6) packet header verification that the flow module for SRX Series devices performs.

To ensure the integrity of an IPv6 packet, the flow module performs the following sanity checks.

For all IPv6 packets, it checks the following parts of the header:

- TCP length
- UDP length
- Hop-by-hop extension to ensure that it follows the basic IPv6 header and does not come after another extension header
- That the IP data length error (IP length—total extension header length is not less than zero (<0))

In addition to these verifications, the software performs other standard checks such as verifying that the correct IP version is specified and that the length of the IP address is correct.

- Related Documentation**
- [About the IPv6 Basic Packet Header on page 142](#)
  - [Understanding IPv6 Packet Header Extensions on page 144](#)
  - [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)

## Understanding Path MTU Messages for IPv6 Packets

**Supported Platforms** [SRX Series, vSRX](#)

This topic 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. If it 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.

#### Related Documentation

- [Understanding How SRX Series Devices Handle ICMPv6 Packets on page 147](#)
- [About the IPv6 Basic Packet Header on page 142](#)
- [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)

## Understanding How SRX Series Devices Handle Packet Fragmentation for IPv6 Flows

**Supported Platforms** [SRX Series, vSRX](#)

This topic explains packet fragmentation for IP version 6 (IPv6).

For IPv4 Internet Control Message Protocol (IPv4 ICMP), 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, only a 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.

#### Related Documentation

- [Understanding How SRX Series Devices Handle ICMPv6 Packets on page 147](#)
- [Understanding Path MTU Messages for IPv6 Packets on page 145](#)
- [Understanding IPv6 Packet Header Extensions on page 144](#)
- [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)

## Understanding How SRX Series Devices Handle ICMPv6 Packets

**Supported Platforms** [SRX Series, vSRX](#)

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 12 on page 148](#) 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 12: ICMPv6 Packets That Junos OS Handles Differently from Other ICMPv6 Packets**

Message	Meaning
01-Destination Unreachable	<p>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.</p> <p>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.</p> <p>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.</p>

**Table 12: ICMPv6 Packets That Junos OS Handles Differently from Other ICMPv6 Packets (*continued*)**

Message	Meaning
02-Packet Too Big	<p>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.</p> <p>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.</p> <ul style="list-style-type: none"> <li>• If a matching session exists, the device delivers it to the source node.</li> <li>• If a matching session does not exist, the device drops the packet</li> </ul> <p><b>NOTE:</b> A Packet Too Big message is the only kind of ICMPv6 packet that will cause the flow module to modify an interface.</p>
03-Time Exceeded	<p>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.</p>
04-Parameter Problem	<p>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.</p>

**Related  
Documentation**

- [Understanding Path MTU Messages for IPv6 Packets on page 145](#)
- [Understanding How SRX Series Devices Handle Packet Fragmentation for IPv6 Flows on page 147](#)
- [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)



## CHAPTER 9

# Configuring IPv6 Dual-Stack

- [Understanding IPv6 Dual-Stack Lite on page 151](#)
- [Example: Configuring IPv6 Dual-Stack Lite on page 154](#)

## Understanding IPv6 Dual-Stack Lite

---

**Supported Platforms** [SRX Series, vSRX](#)

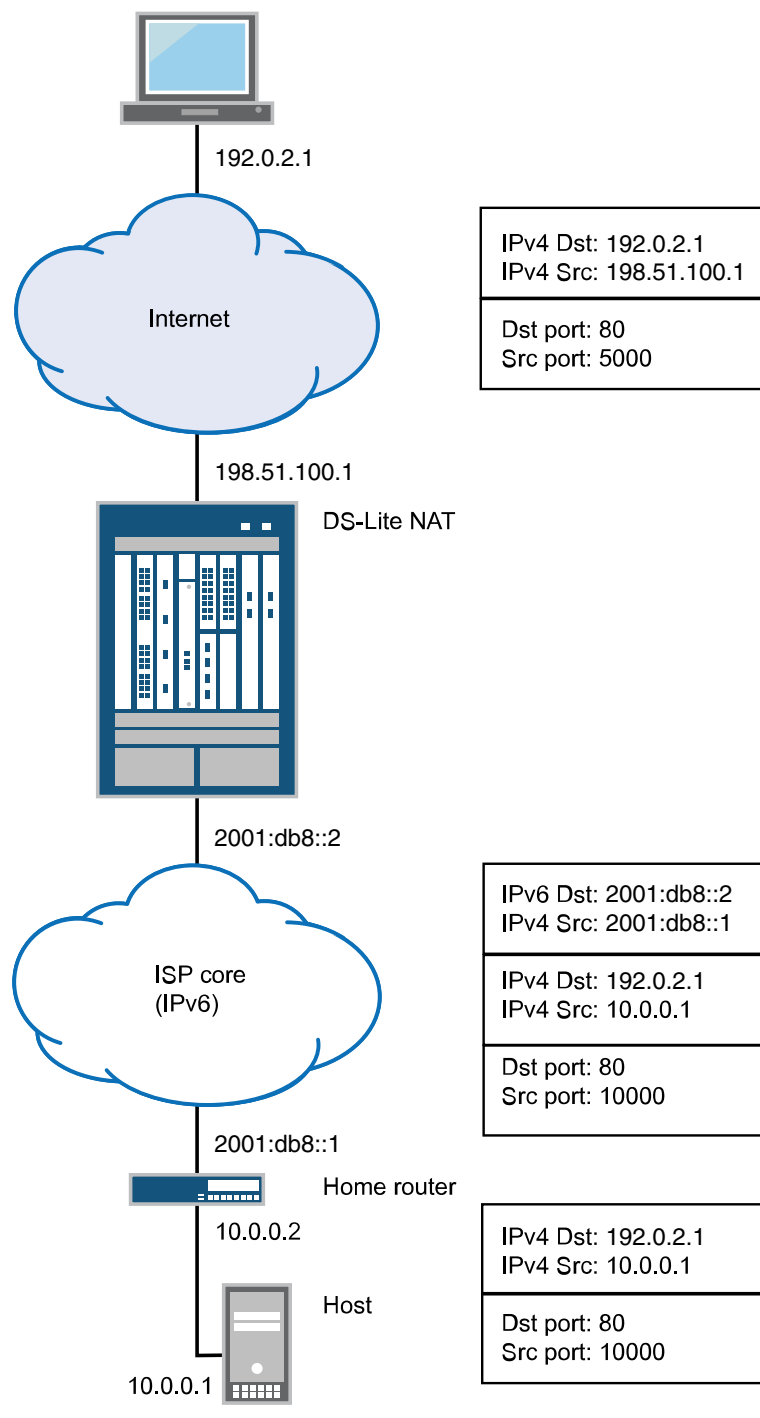
IPv6 dual-stack lite (DS-Lite) is a technology that enables Internet service providers to move to an IPv6 network while simultaneously handling IPv4 address depletion.

IPv4 addresses are becoming depleted; therefore, broadband service providers (DSL, cable, and mobile) need new addresses to support new users. Providing IPv6 addresses alone is often not workable because most of the systems that make up the public Internet are still enabled and support only IPv4, and many users' systems do not yet fully support IPv6.

DS-Lite allows service providers to migrate to an IPv6 access network without changing end-user software. The device that accesses the Internet remains the same, thus allowing IPv4 users to continue accessing IPv4 internet content with minimum disruption to their home networks, while enabling IPv6 users to access IPv6 content.

[Figure 10 on page 152](#) illustrates the DS-Lite architecture which uses IPv6-only links between the provider and the user while maintaining the IPv4 (or dual-stack) hosts in the user network.

Figure 10: DS-Lite NAT (IPv4-in-IPv6)



The DS-Lite deployment model consists of the following components:



- Software initiator for the DS-Lite home router--Encapsulates the IPv4 packet and transmits it across an IPv6 tunnel.
- Software concentrator for DS-Lite carrier-grade Network Address Translation (NAT)--Decapsulates the IPv4-in-IPv6 packet and also performs IPv4-IPv4 NAT translations.

When a user's device sends an IPv4 packet to an external destination, DS-Lite encapsulates the IPv4 packet in an IPv6 packet for transport into the provider network. These IPv4-in-IPv6 tunnels are called *softwires*. Tunneling IPv4 over IPv6 is simpler than translation and eliminates performance and redundancy concerns.

The softwires terminate in a software concentrator at some point in the service provider network, which decapsulates the IPv4 packets and sends them through a carrier-grade Network Address Translation (NAT) device. There, the packets undergo source NAT processing to hide the original source address.

IPv6 packets originated by hosts in the subscriber's home network are transported natively over the access network.

The DS-Lite carrier-grade NAT translates IPv4-to-IPv4 addresses to multiple subscribers through a single global IPv4 address. Overlapping address spaces used by subscribers are disambiguated through the identification of tunnel endpoints. One concentrator can be the endpoint of multiple softwires.

The IPv4 packets originated by the end hosts have private (and possibly overlapping) IP addresses. Therefore, NAT must be applied to these packets. If end hosts have overlapping addresses, Network Address Port Translation (NAPT) is needed.

Using NAPT, the system adds the source address of the encapsulating IPv6 packet in the subscriber network to the inside IPv4 source address and port. Because each user's IPv6 address is unique, the combination of the IPv6 source address with the IPv4 source address and port creates an unambiguous mapping.

The system takes the following actions when it receives a responding IPv4 packet from outside the subscriber network:

- Encapsulates the IPv4 packet in an IPv6 packet using the mapped IPv6 address as the IPv6 destination address.
- Forwards the packet to the user.

[Table 13 on page 153](#) lists the maximum number of software initiators and software concentrators per device. Platform support depends on the Junos OS release in your installation.

**Table 13: Software Initiator and Software Concentrator Capacity**

Description	SRX650	SRX1500	SRX3400 SRX3600	SRX5400 SRX5600 SRX5800
Maximum software initiators connected per device	50,000	300	100,000	100,000

Table 13: Software Initiator and Software Concentrator Capacity (*continued*)

Maximum software concentrator numbers per device	32	32	32	32
--	----	----	----	----



**NOTE:** The most recent IETF draft documentation for DS-Lite uses new terminology:

- The term software initiator has been replaced by B4.
- The term software concentrator has been replaced by AFTR.

Junos OS documentation generally uses the original terms when discussing configuration in order to be consistent with the CLI statements used to configure DS-Lite.

For more information, see the following documents:

- draft-ietf-software-dual-stack-lite-06, *Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion*, August 2010.
- RFC 2473, *Generic Packet Tunneling in IPv6 Specification*, December 1998.
- RFC 2663, *IP Network Address Translator (NAT) Terminology and Considerations*, August 1999.
- RFC 4787, *Network Address Translation (NAT) Behavioral Requirements for Unicast UDP, BCP 127*, January 2007.
- RFC 4925, *Software Problem Statement*, July 2007.
- RFC 5382, *NAT Behavioral Requirements for TCP, BCP 142*, October 2008.
- RFC 5508, *NAT Behavioral Requirements for ICMP, BCP 148*, April 2009.
- <http://www.potaroo.net/tools/ipv4/index.html>
- <http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xhtml>

#### Related Documentation

- [Example: Configuring IPv6 Dual-Stack Lite on page 154](#)
- [Understanding IPv6 Address Space, Addressing, Address Format, and Address Types](#)
- [Understanding How SRX Series Devices Handle ICMPv6 Packets on page 147](#)
- [About the IPv6 Basic Packet Header on page 142](#)

## Example: Configuring IPv6 Dual-Stack Lite

**Supported Platforms** [SRX Series, vSRX](#)

When an ISP begins to allocate IPv6 addresses and IPv6-capable equipment to new subscriber homes, dual-stack lite (DS-Lite) provides a method for the private IPv4 addresses behind the IPv6 CE WAN equipment to reach the IPv4 network. DS-Lite enables

IPv4 customers to continue to access the Internet using their current hardware by using a softwire initiator at the customer edge to encapsulate IPv4 packets into IPv6 packets with minimum disruption to their home network, while enabling IPv6 customers to access IPv6 content. The softwire concentrator decapsulates the IPv4-in-IPv6 packets and also performs IPv4-IPv4 NAT translations.

This example shows you how to configure a softwire concentrator for IPv4-in-IPv6 addresses.

- [Requirements on page 155](#)
- [Overview on page 155](#)
- [Configuration on page 155](#)
- [Verification on page 156](#)

## Requirements

Before you begin:

- Review the overview section on DS-Lite. See “[Understanding IPv6 Dual-Stack Lite](#)” on [page 151](#).
- Review how ICMPv6 packets are handled by the SRX Series devices. See “[Understanding How SRX Series Devices Handle ICMPv6 Packets](#)” on [page 147](#).

## Overview

This configuration example shows how to configure a softwire concentrator, the softwire name, the concentrator address, and the softwire type.



**NOTE:** The softwire concentrator IPv6 address can match an IPv6 address configured on a physical interface or an IPv6 address configured on a loopback interface.

## 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 security softwires softwire-name my_sc1 softwire-concentrator 2001:db8::1
  softwire-type IPv4-in-IPv6
```

**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 DS-Lite software concentrator to convert IPv4 packets into IPv6 packets:

1. Assign a name for the software concentrator.

```
[edit security]
user@host# edit softwires software-name my_sc1
```

2. Specify the address of the software concentrator.

```
[edit security softwires software-name my_sc1]
user@host# set software-concentrator 2001:db8::1
```

3. Specify the software type for IPv4 to IPv6.

```
[edit security softwires software-name my_sc1 software-concentrator 2001:db8::1]
user@host# set software-type IPv4-in-IPv6
```

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

```
[edit security softwires software-name my_sc1]
user@host# show
software-concentrator 2001:db8::1;
software-type ipv4-in-ipv6;
```

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

## Verification

From operational mode, enter the **show security softwires** command. If a software is not connected, the operational output looks like the following sample:

```
user@host# show security softwires
Software Name      SC Address      Status  Number of SI connected
my-sc1            2001:db8::1     Active  0
```

If a software is connected, the operational output looks like the following sample:

```
user@host# show security softwires
Software Name      SC Address      Status  Number of SI connected
my-sc1            2001:db8::1     Connected  1
```

- Related Documentation**
- [Understanding IPv6 Dual-Stack Lite on page 151](#)
  - *Understanding IPv6 Address Space, Addressing, Address Format, and Address Types*
  - [About the IPv6 Basic Packet Header on page 142](#)



## PART 5

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

- [Monitoring Security Flow Sessions on page 161](#)
- [Monitoring X2 Traffic By Configuring Mirror Filters on page 177](#)





## CHAPTER 10

# Monitoring Security Flow Sessions

- [Monitoring Security Flow Sessions Overview on page 161](#)
- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
- [Displaying Global Session Parameters for All SRX Series Services Gateways on page 164](#)
- [Displaying a Summary of Sessions for SRX Series Services Gateways on page 165](#)
- [Displaying Session and Flow Information About Sessions for SRX Series Services Gateways on page 166](#)
- [Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways on page 166](#)
- [Using Filters to Display Session and Flow Information for SRX Series Services Gateways on page 167](#)
- [Information Provided in Session Log Entries for SRX Series Services Gateways on page 167](#)
- [Error Handling Extensions on page 172](#)

## Monitoring Security Flow Sessions Overview

---

### Supported Platforms [SRX Series, vSRX](#)

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.

#### Related Documentation

- [monitor security flow start on page 331](#)
- [monitor security flow file on page 327](#)
- [monitor security flow filter on page 329](#)
- [show monitor security flow on page 421](#)
- [monitor security flow stop on page 332](#)
- [clear monitor security flow filter on page 301](#)

---

## Understanding How to Obtain Session Information for SRX Series Services Gateways

Supported Platforms [SRX Series, vSRX](#)

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. See [“Information Provided in Session Log Entries for SRX Series Services Gateways” on page 167](#) for details about session information provided in system logs.

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
```

#### Related Documentation

- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)
- [Clearing Sessions for SRX Series Services Gateways on page 50](#)
- [Displaying Global Session Parameters for All SRX Series Services Gateways on page 164](#)
- [allow-embedded-icmp on page 221](#)

## Displaying Global Session Parameters for All SRX Series Services Gateways

**Supported Platforms** [SRX Series, vSRX](#)

**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.

**Related Documentation**

- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
- [Displaying a Summary of Sessions for SRX Series Services Gateways on page 165](#)
- [Displaying Session and Flow Information About Sessions for SRX Series Services Gateways on page 166](#)
- [Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways on page 166](#)
- [Using Filters to Display Session and Flow Information for SRX Series Services Gateways on page 167](#)
- [Information Provided in Session Log Entries for SRX Series Services Gateways on page 167](#)

## Displaying a Summary of Sessions for SRX Series Services Gateways

**Supported Platforms** [SRX Series, vSRX](#)

**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
```

**Related Documentation**

- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
- [Displaying Global Session Parameters for All SRX Series Services Gateways on page 164](#)
- [Displaying Session and Flow Information About Sessions for SRX Series Services Gateways on page 166](#)
- [Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways on page 166](#)
- [Using Filters to Display Session and Flow Information for SRX Series Services Gateways on page 167](#)

- [Information Provided in Session Log Entries for SRX Series Services Gateways on page 167](#)

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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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
```

- Related Documentation**
- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
  - [Displaying Global Session Parameters for All SRX Series Services Gateways on page 164](#)
  - [Displaying a Summary of Sessions for SRX Series Services Gateways on page 165](#)
  - [Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways on page 166](#)
  - [Using Filters to Display Session and Flow Information for SRX Series Services Gateways on page 167](#)
  - [Information Provided in Session Log Entries for SRX Series Services Gateways on page 167](#)

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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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
```

- Related Documentation**
- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
  - [Displaying Global Session Parameters for All SRX Series Services Gateways on page 164](#)
  - [Displaying a Summary of Sessions for SRX Series Services Gateways on page 165](#)
  - [Displaying Session and Flow Information About Sessions for SRX Series Services Gateways on page 166](#)
  - [Using Filters to Display Session and Flow Information for SRX Series Services Gateways on page 167](#)
  - [Information Provided in Session Log Entries for SRX Series Services Gateways on page 167](#)

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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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
```

- Related Documentation**
- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
  - [Displaying Global Session Parameters for All SRX Series Services Gateways on page 164](#)
  - [Displaying a Summary of Sessions for SRX Series Services Gateways on page 165](#)
  - [Displaying Session and Flow Information About Sessions for SRX Series Services Gateways on page 166](#)
  - [Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways on page 166](#)
  - [Information Provided in Session Log Entries for SRX Series Services Gateways on page 167](#)

## Information Provided in Session Log Entries for SRX Series Services Gateways

---

**Supported Platforms** [SRX Series, vSRX](#)

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 14 on page 168](#), [Table 15 on page 169](#), and [Table 16 on page 171](#). 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 14: Session Create Log Fields**

Field	Description
<b>source-address</b>	Source IP address of the packet that created the session.
<b>source-port</b>	Source port of the packet that created the session.
<b>destination-address</b>	Destination IP address of the packet that created the session.
<b>destination-port</b>	Destination port of the packet that created the session.
<b>service-name</b>	Application that the packet traversed (for example, "junos-telnet" for Telnet traffic during the session allowed by a policy that permits native Telnet).
<b>nat-source-address</b>	The translated NAT source address if NAT was applied; otherwise, the source address as above.
<b>nat-source-port</b>	The translated NAT source port if NAT was applied; otherwise, the source port as above.
<b>nat-destination-address</b>	The translated NAT destination address if NAT was applied; otherwise, the destination address as above.
<b>nat-destination-port</b>	The translated NAT destination port if NAT was applied; otherwise, the destination port as above.
<b>src-nat-rule-name</b>	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.*
<b>dst-nat-rule-name</b>	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.*
<b>protocol-id</b>	The protocol ID of the packet that created the session.
<b>policy-name</b>	The name of the policy that permitted the session creation.
<b>session-id-32</b>	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` fields are introduced in the NAT rule session.

**Table 15: Session Close Log Fields**

Field	Description
<code>reason</code>	The reason the session was closed.
<code>source-address</code>	Source IP address of the packet that created the session.
<code>source-port</code>	Source port of the packet that created the session.
<code>destination-address</code>	Destination IP address of the packet that created the session.
<code>destination-port</code>	Destination port of the packet that created the session.
<code>service-name</code>	Application that the packet traversed (for example, "junos-telnet" for Telnet traffic during the session allowed by a policy that permits native Telnet).
<code>nat-source-address</code>	The translated NAT source address if NAT was applied; otherwise, the source address as above.
<code>nat-source-port</code>	The translated NAT source port if NAT was applied; otherwise, the source port as above.
<code>nat-destination-address</code>	The translated NAT destination address if NAT was applied; otherwise, the destination address as above.
<code>nat-destination-port</code>	The translated NAT destination port if NAT was applied; otherwise, the destination port as above.
<code>src-nat-rule-name</code>	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.*
<code>dst-nat-rule-name</code>	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.*
<code>protocol-id</code>	The protocol ID of the packet that created the session.
<code>policy-name</code>	The name of the policy that permitted the session creation.
<code>session-id-32</code>	The 32-bit session ID.
<code>packets-from-client</code>	The number of packets sent by the client related to this session.

Table 15: Session Close Log Fields (*continued*)

Field	Description
<b>bytes-from-client</b>	The number of data bytes sent by the client related to this session.
<b>packets-from-server</b>	The number of packets sent by the server related to this session.
<b>bytes-from-server</b>	The number of data bytes sent by the server related to this session.
<b>elapsed-time</b>	The total session elapsed time from permit to close, given in seconds.
<b>unset</b>	<p>During the session creation, you can set the session close reason as <b>unset</b>.</p> <p>The session closes with the reason <b>unset</b> 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.</p>
<b>TCP CLIENT RST</b>	The session was closed by a TCP reset packet sent to it from the client.
<b>TCP SERVER RST</b>	The session was closed by a TCP reset packet sent to it from the server.
<b>TCP FIN</b>	FIN received from either end.
<b>response received</b>	Response received for a packet request (for example, ICMP req-reply).
<b>ICMP error</b>	ICMP error received.
<b>aged out</b>	Session aged out was reached.
<b>ALG</b>	ALG errors closed the session (for example, remote access server (RAS) maximum limit reached).
<b>HA</b>	HA message closed the session.
<b>idle Timeout</b>	There was no traffic for the session before the configured age-out time was reached.
<b>auth</b>	Authentication failed.
<b>IDP</b>	IDP closed the session because of security module (SM) internal error.
<b>synproxy failure</b>	SYN proxy failure closed the session.

Table 15: Session Close Log Fields (*continued*)

Field	Description
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.
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 16: 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.

Table 16: Session Deny Log Fields (*continued*)

Field	Description
<b>destination-port</b>	Destination port of the packet that attempted to create the session.
<b>service-name</b>	Application that the packet attempted to traverse.
<b>protocol-id</b>	The protocol ID of the packet that attempted to create the session.
<b>icmp-type</b>	The ICMP type if the denied packet was ICMP configured; otherwise, this field will be 0.
<b>policy-name</b>	The name of the policy that denied the session creation.

**Release History Table**

Release	Description
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**

- [Understanding How to Obtain Session Information for SRX Series Services Gateways on page 162](#)
- [Displaying Global Session Parameters for All SRX Series Services Gateways on page 164](#)
- [Displaying a Summary of Sessions for SRX Series Services Gateways on page 165](#)
- [Displaying Session and Flow Information About Sessions for SRX Series Services Gateways on page 166](#)
- [Displaying Session and Flow Information About a Specific Session for SRX Series Services Gateways on page 166](#)
- [Clearing Sessions for SRX Series Services Gateways on page 50](#)
- [Using Filters to Display Session and Flow Information for SRX Series Services Gateways on page 167](#)

**Error Handling Extensions****Supported Platforms** SRX Series, vSRX

- [Understanding Chassis Manager FPC Fault Detection and Error Handling Enhancements on page 173](#)

## Understanding Chassis Manager FPC Fault Detection and Error Handling Enhancements

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 on page 173](#)
- [Error Detection and Management on page 173](#)
- [Error Detection Processes on page 174](#)
- [Integration with Chassis Cluster on page 175](#)
- [Wedge Detection, Reporting, and Management on page 175](#)

### 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.

- See Also**
- [error on page 231](#)
  - [show chassis fpc errors on page 345](#)





## CHAPTER 11

# Monitoring X2 Traffic By Configuring Mirror Filters

- [Understanding X2 Traffic Monitoring on page 177](#)
- [Example: Configuring a Mirror Filter for X2 Traffic Monitoring on page 180](#)

## Understanding X2 Traffic Monitoring

---

**Supported Platforms**   [SRX Series, vSRX](#)

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.

This topic covers X2 traffic monitoring on SRX Series devices:

- [X2 Traffic Monitoring Overview on page 177](#)
- [Limitations of X2 Traffic Monitoring on page 179](#)
- [X2 Traffic Terminology on page 179](#)

## 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 11 on page 178](#) 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 11: SRX Series Device in an LTE Mobile Network

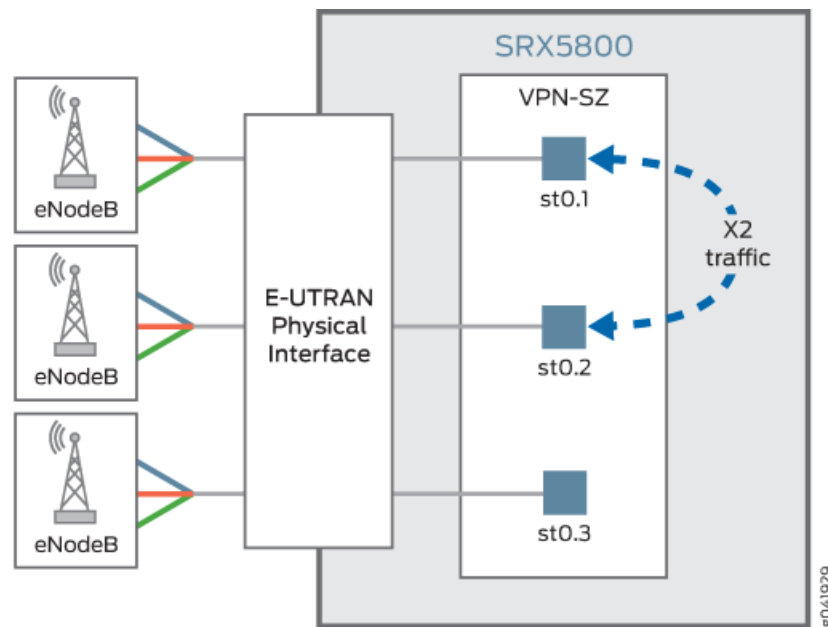
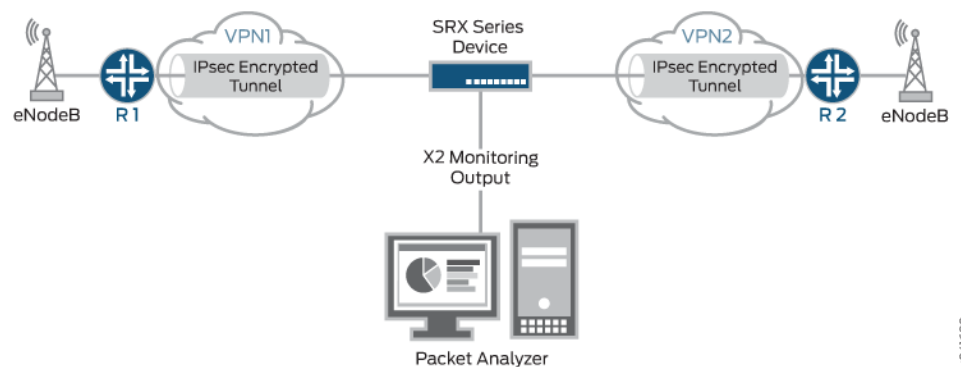


Figure 12 on page 178 shows a mobile operators network with an SRX Series device providing IPsec tunnel connection between the two eNodeBs. The SRX Series device is connected to a packet analyzer (also called a *sniffing* device) that is used for collecting and monitoring the X2 traffic. The IPsec tunnel 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 12: 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.



**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 17 on page 179 lists some X2 traffic related terms and their descriptions.

**Table 17: 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.
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: <ul style="list-style-type: none"> <li>• Manages mobility and load</li> <li>• Reports general error situations</li> <li>• Sets and resets the X2 interface</li> <li>• Updates the eNodeB configuration</li> </ul>

- Related Documentation**
- [Example: Configuring a Mirror Filter for X2 Traffic Monitoring on page 180](#)
  - [mirror-filter \(Security Forwarding Options\) on page 258](#)

---

## Example: Configuring a Mirror Filter for X2 Traffic Monitoring

**Supported Platforms** [SRX Series, vSRX](#)

This example shows how to configure a mirror filter to monitor X2 traffic between two eNodeBs in an LTE mobile network.

- [Requirements on page 180](#)
- [Overview on page 180](#)
- [Configuration on page 182](#)
- [Verification on page 183](#)

### Requirements

Before you begin:

- Understand X2 traffic monitoring. See [“Understanding X2 Traffic Monitoring” on page 177](#).
- 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. See *Example: Configuring a Route-Based VPN*.

### 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:

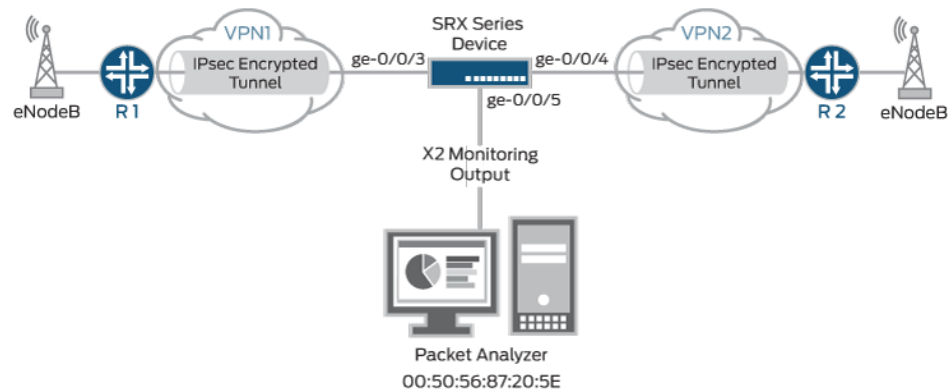
- destination IP address prefix
- destination port
- IP protocol
- source IP address prefix
- source port
- incoming and outgoing interfaces

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 13 on page 181 shows the SRX Series device connecting to the eNodeBs using IPsec tunnels. The SRX Series device is also connected to a packet analyzer.

**Figure 13: 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.

**Related Documentation**

- [mirror-filter \(Security Forwarding Options\) on page 258](#)
- [clear security forward-options mirror filter on page 326](#)





## PART 6

# Configuring Packet-Based Forwarding

- [Configuring Selective Stateless Packet-Based Services on page 187](#)



## CHAPTER 12

# Configuring Selective Stateless Packet-Based Services

- [Understanding Packet-Based Processing on page 187](#)
- [Understanding Selective Stateless Packet-Based Services on page 188](#)
- [Selective Stateless Packet-Based Services Configuration Overview on page 190](#)
- [Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding on page 192](#)
- [Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding on page 203](#)

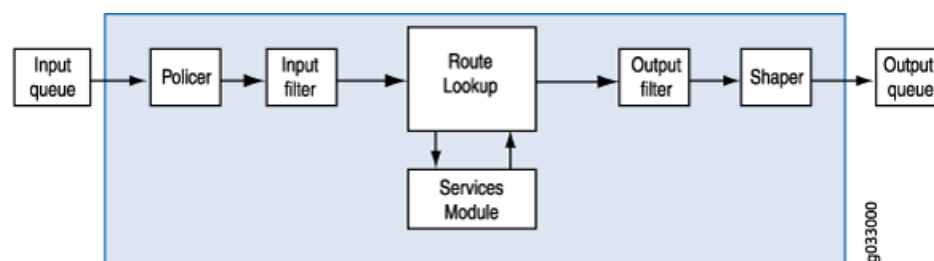
## Understanding Packet-Based Processing

**Supported Platforms** SRX Series, vSRX

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 14 on page 187](#) shows the traffic flow for packet-based forwarding.

**Figure 14: 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."

**Related Documentation**

- [Understanding Selective Stateless Packet-Based Services on page 188](#)
- [Selective Stateless Packet-Based Services Configuration Overview on page 190](#)
- [Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding on page 192](#)
- [Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding on page 203](#)
- *forwarding-options (Security)*

---

## Understanding Selective Stateless Packet-Based Services

---

**Supported Platforms**   [SRX Series, vSRX](#)

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.

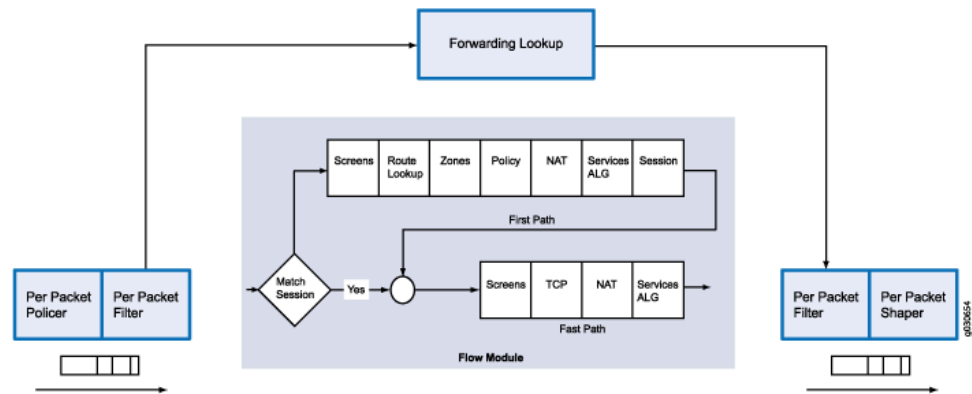
By default, Juniper Networks Security devices running Junos OS use flow-based forwarding. Selective stateless packet-based services allows you to configure the device to provide only packet-based processing for selected traffic based on input filter terms. Other traffic is processed for flow-based forwarding. 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 15 on page 189 shows traffic flow with selective stateless packet-based services bypassing flow-based processing.

Figure 15: 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.

**Related Documentation**

- [Understanding Packet-Based Processing on page 187](#)
- [Selective Stateless Packet-Based Services Configuration Overview on page 190](#)
- [Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding on page 192](#)
- [Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding on page 203](#)

---

## Selective Stateless Packet-Based Services Configuration Overview

---

**Supported Platforms**    [SRX1500, SRX300, SRX320, SRX340, SRX345, vSRX](#)

This feature is supported on SRX 300, SRX 320, SRX 340, SRX 345, SRX 550M, and SRX 1500 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

**Related Documentation**

- [Understanding Packet-Based Processing on page 187](#)
- [Understanding Selective Stateless Packet-Based Services on page 188](#)
- [Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding on page 192](#)
- [Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding on page 203](#)

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

---

**Supported Platforms** [SRX Series, vSRX](#)

This example shows how to configure selective stateless packet-based services for end-to-end packet-based forwarding. This feature is supported on the SRX 300, SRX 320, SRX 340, SRX 345, 550M, and SRX 1500 devices

- [Requirements on page 192](#)
- [Overview on page 193](#)
- [Configuration on page 194](#)
- [Verification on page 200](#)

### Requirements

Before you begin:

- Understand how to configure stateless firewall filters. See the [\[Warning: element unresolved in stylesheets: <subordinate-index> \(in <para>\)\]. This is probably a new element that is not yet supported in the stylesheets.](#) [\[Warning: element unresolved in stylesheets: <title> \(in <subordinate-index>\)\]. This is probably a new element that is not yet supported in the stylesheets.](#) Junos OS Routing Protocols Library

**Supported Platforms** [M Series, MX Series, PTX Series, SRX Series, T Series](#)



Starting in February 2017, Junos OS documentation is release independent. This means that for any given guide (except release notes), you can find all of the content for all supported releases in one place.

To determine if a feature is supported by a specific platform or Junos OS release, refer to the Junos OS release notes, [Feature Explorer](#), the new Release History tables in the documentation, and the release information in the CLI content.

These guides provide an overview of general routing protocol concepts and configuration information, including multitopology routing, interior gateway protocols (IS-IS, OSPF, RIP), and BGP. They also provide configuration examples and reference information for each protocol.

[Warning: element unresolved in stylesheets: <category> (in <subordinate-index>). This is probably a new element that is not yet supported in the stylesheets.]

[Warning: element unresolved in stylesheets: <title> (in <category>). This is probably a new element that is not yet supported in the stylesheets.]

Routing Configuration

[Warning: element unresolved in stylesheets: <link-list> (in <category>). This is probably a new element that is not yet supported in the stylesheets.]

*Routing Protocols Overview*  
*BGP Feature Guide*  
*CMP Router Discovery Protocol Feature Guide*  
*IS-IS Feature Guide*  
*Multitopology Routing Feature Guide*  
*IPv6 Neighbor Discovery Feature Guide*  
*OSPF Feature Guide*  
*Protocol-Independent Routing Properties Feature Guide*  
*RIP Feature Guide*  
*RIPng Feature Guide*

[Warning: element unresolved in stylesheets: <promo-box> (in <subordinate-index>). This is probably a new element that is not yet supported in the stylesheets.]

#### Related Documentation

- [FAQ: Routing Protocol Process \(rpd\)](#)
- [Junos OS Routing Protocols Library for Routing Devices](#)
- 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.

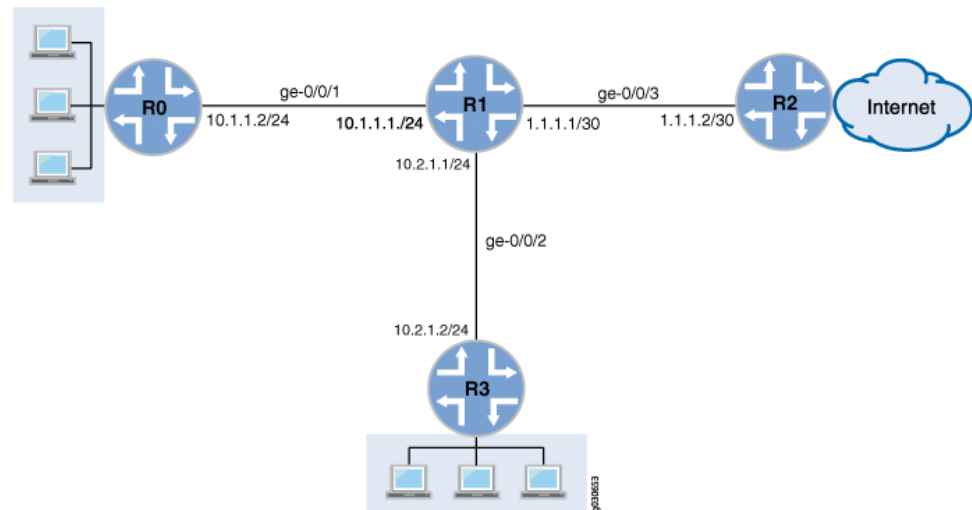
Then on device R1 you configure a zone called untrust and assign it to interface ge-0/0/3. You also create a zone called trust and assign interfaces ge-0/0/1 and ge-0/0/2 to it. You configure trust and untrust zones to allow all supported application services as inbound services. You allow traffic from 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 16 on page 194 shows the network topology used in this example.

Figure 16: 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 description "Internal 1" ge-0/0/1 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 description "Internal 1" ge-0/0/1 unit 0 family inet address 10.1.1.1/24
set interfaces description "Internal 2" ge-0/0/2 unit 0 family inet address 10.2.1.1/24
set interfaces description "Internet" ge-0/0/3 unit 0 family inet address 203.0.113.1/30
set routing-options static route 0.0.0.0/0 next-hop 203.0.113.2/30
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 description "Internal 1" ge-0/0/1 unit 0 family inet filter input
  bypass-flow-filter
set interfaces description "Internal 2" ge-0/0/2 unit 0 family inet filter input
  bypass-flow-filter

{device R2}
set interfaces description "Internet" ge-0/0/3 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 description "Internal 2" ge-0/0/2 unit 0 family inet address 10.2.1.2/24
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 description "Internal 1" ge-0/0/1 unit 0 family inet address
  10.1.1.2/24

{device R1}
[edit]
user@host# set interfaces description "Internal 1" ge-0/0/1 unit 0 family inet address
  10.1.1.1/24
user@host# set interfaces description "Internal 2" ge-0/0/2 unit 0 family inet address
  10.2.1.1/24
user@host# set interfaces description "Internet" ge-0/0/3 unit 0 family inet address
  203.0.113.1/30

```

```
{device R2}
[edit]
user@host# set interfaces description "Internet" ge-0/0/3 unit 0 family inet address
203.0.113.2/30

{device R3}
[edit]
user@host# set interfaces description "Internal 2" ge-0/0/2 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 description "Internal 1" ge-0/0/1 unit 0 family inet filter
input bypass-flow-filter
user@host# set interfaces description "Internal 2" ge-0/0/2 unit 0 family inet filter
input bypass-flow-filter

```

**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

Confirm that the configuration is working properly.

- [Verifying the End-to-End Packet-Based Configuration on page 200](#)
- [Verifying Session Establishment on Intranet Traffic on page 200](#)
- [Verifying Session Establishment on Internet Traffic on page 202](#)

---

### 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 RO}
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.

Release History Table

Release	Description
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.

### 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.

Release History Table

Release	Description
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.

- Related Documentation
- [Understanding Selective Stateless Packet-Based Services on page 188](#)
  - [Selective Stateless Packet-Based Services Configuration Overview on page 190](#)
  - [Example: Configuring Selective Stateless Packet-Based Services for Packet-Based to Flow-Based Forwarding on page 203](#)

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

Supported Platforms [SRX Series, vSRX](#)

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 on page 204](#)
- [Overview on page 204](#)
- [Configuration on page 205](#)
- [Verification on page 211](#)

## Requirements

Before you begin:

- Understand how to configure stateless firewall filters. See [“Junos OS Routing Protocols Library” on page ?](#).
- Establish basic connectivity. .

## Overview

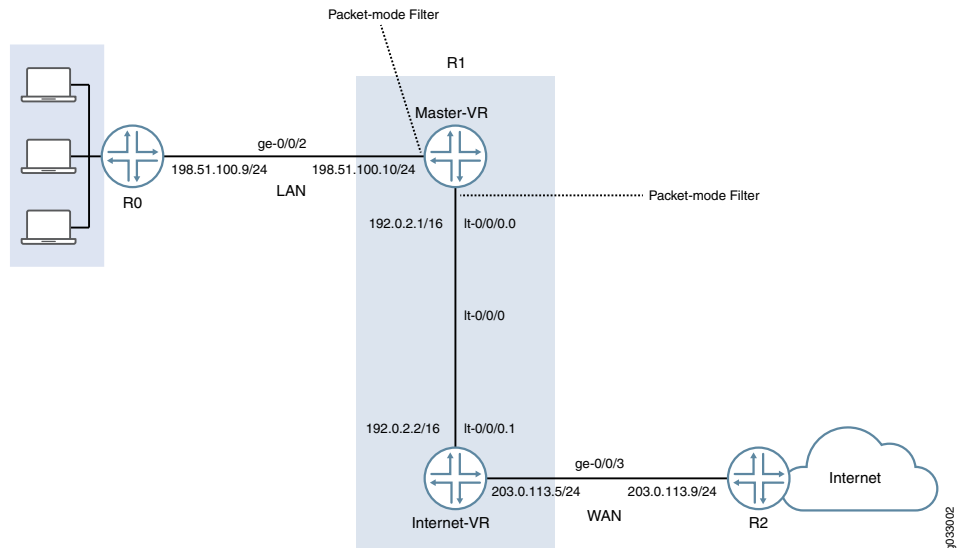
In this example, you configure the IP addresses for the interfaces on each of the devices. For device R0 as 198.51.100.9/24; for R1 the are 198.51.100.10/24 and 203.0.113.5/24; and for R2 it is 203.0.113.9/24. On device R1, you set an internal service interface lt-0/0/0 between routing instances and 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 17 on page 205](#) shows the network topology used in this example.

Figure 17: 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;
        }
      }
    }
    interfaces {
      ge-0/0/2.0;
      lt-0/0/0.0;
    }
  }
}
security zone Internet-VR-zone {
  host-inbound-traffic {
    system-services {
      all;
      {
        protocols {
          all;
        }
      }
    }
    interfaces {
      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

Confirm that the configuration is working properly.

- [Verifying the Packet-Based to Flow-Based Configuration on page 211](#)
- [Verifying Session Establishment on LAN Traffic on page 211](#)
- [Verifying Session Establishment on Internet Traffic on page 212](#)

---

### 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**.

#### Related Documentation

- [Understanding Selective Stateless Packet-Based Services on page 188](#)
- [Selective Stateless Packet-Based Services Configuration Overview on page 190](#)

- [Example: Configuring Selective Stateless Packet-Based Services for End-to-End Packet-Based Forwarding on page 192](#)

## PART 7

# Configuration Statements and Operational Commands

- [Configuration Statements on page 217](#)
- [Operational Commands on page 297](#)





## CHAPTER 13

# Configuration Statements

- [aging](#) on page 219
- [all-tcp](#) on page 220
- [allow-dns-reply](#) on page 220
- [allow-embedded-icmp](#) on page 221
- [allow-reverse-ecmp](#) on page 221
- [application-services \(Security Forwarding Process\)](#) on page 222
- [apply-to-half-close-state](#) on page 223
- [ethernet-switching](#) on page 224
- [destination-header](#) on page 225
- [destination-port \(Security Forwarding Options\)](#) on page 226
- [destination-prefix \(Security Forwarding Options\)](#) on page 230
- [early-ageout](#) on page 230
- [error](#) on page 231
- [fin-invalidate-session](#) on page 234
- [flow \(Security Flow\)](#) on page 236
- [force-ip-reassembly](#) on page 238
- [forwarding-process](#) on page 239
- [fpc error](#) on page 241
- [fru-poweron-sequence](#) on page 243
- [gre-in](#) on page 244
- [gre-out](#) on page 245
- [high-watermark](#) on page 246
- [hop-by-hop-header](#) on page 247
- [icmpv6-malformed](#) on page 248
- [idle-timeout \(System Services\)](#) on page 249
- [inline-tap](#) on page 250
- [interface-in \(Security Forwarding Options\)](#) on page 250
- [interface-out \(Security Forwarding Options\)](#) on page 251

- [ipv4-template \(Services\)](#) on page 251
- [ipv6-extension-header](#) on page 252
- [ipv6-extension-header-limit](#) on page 253
- [ipv6-malformed-header](#) on page 254
- [ipv6-template \(Services\)](#) on page 254
- [low-latency](#) on page 255
- [low-watermark](#) on page 256
- [maximize-idp-sessions](#) on page 257
- [mirror-filter \(Security Forwarding Options\)](#) on page 258
- [mode \(Security Forwarding Options\)](#) on page 260
- [no-sequence-check](#) on page 261
- [no-tcp-reset](#) on page 262
- [output \(Security Forwarding Options\)](#) on page 263
- [packet-filter](#) on page 264
- [packet-log \(Security Flow\)](#) on page 265
- [packet-ordering-mode \(Application Services\)](#) on page 266
- [pending-sess-queue-length](#) on page 267
- [per-session-statistics](#) on page 267
- [preserve-incoming-fragment-size](#) on page 268
- [propagate-settings](#) on page 269
- [protocol \(Security Forwarding Options\)](#) on page 270
- [resource-manager](#) on page 271
- [route-change-timeout](#) on page 272
- [rst-invalidate-session](#) on page 273
- [rst-sequence-check](#) on page 274
- [sampling](#) on page 275
- [services-offload](#) on page 276
- [session \(System Services\)](#) on page 277
- [session-limit \(System Services\)](#) on page 278
- [np-cache \(Flexible PIC Concentrator\)](#) on page 279
- [source-port \(Security Forwarding Options\)](#) on page 280
- [source-prefix \(Security Forwarding Options\)](#) on page 284
- [syn-flood-protection-mode](#) on page 284
- [tcp-initial-timeout](#) on page 285
- [tcp-mss \(Security Flow\)](#) on page 286
- [tcp-session](#) on page 287
- [time-wait-state](#) on page 288

- [traceoptions \(Security\)](#) on page 289
- [traceoptions \(Security Flow\)](#) on page 291
- [transport \(Security Log\)](#) on page 295
- [weight \(Security\)](#) on page 296

## aging

**Supported Platforms** [SRX Series, vSRX](#)

**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** Direct the device to begin aggressively aging out sessions when the percentage of entries in the session table exceeds the high-watermark setting and then stops when the percentage of sessions 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**

- [Juniper Networks Devices Processing Overview](#) on page 3

## all-tcp

---

Supported Platforms	<a href="#">SRX Series, vSRX</a>
Syntax	all-tcp mss <i>value</i> ;
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	<b>mss <i>value</i></b> —TCP MSS value. <b>Range:</b> 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	<ul style="list-style-type: none"><li>• <a href="#">tcp-mss (Security Flow) on page 286</a></li><li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li></ul>

## allow-dns-reply

---

Supported Platforms	<a href="#">SRX Series, vSRX</a>
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 <b>allow-dns-reply</b> 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	<ul style="list-style-type: none"><li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li></ul>

## allow-embedded-icmp

---

<b>Supported Platforms</b>	<a href="#">SRX Series, vSRX</a>
<b>Syntax</b>	allow-embedded-icmp;
<b>Hierarchy Level</b>	[edit security flow]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.3X48-D10.
<b>Description</b>	Allow an embedded ICMP packet to pass through the device even when there is no session match. 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.
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li> </ul>

## allow-reverse-ecmp

---

<b>Supported Platforms</b>	<a href="#">SRX Series</a>
<b>Syntax</b>	allow-reverse-ecmp
<b>Hierarchy Level</b>	[edit security flow]
<b>Release Information</b>	Statement introduced in Junos OS Release 17.3.
<b>Description</b>	<p>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.</p> <p>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.</p>
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Understanding ECMP Flow-Based Forwarding on page 53</a></li> <li>• <a href="#">Understanding ECMP Flow-Based Forwarding for Reverse Traffic on SRX Series Devices and vSRX Instances on page 55</a></li> </ul>

## application-services (Security Forwarding Process)

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax**

```
application-services {
  enable-gtpu-distribution;
  maximize-alg-sessions;
  maximize-idp-sessions {
    weight (equal | firewall | idp);
  }
  packet-ordering-mode {
    (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 increase the capacity of IDP processing with the **maximize-idp-sessions** option. 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.

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.

**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** • [Juniper Networks Devices Processing Overview on page 3](#)

---

## apply-to-half-close-state

---

**Supported Platforms** [SRX Series, vSRX](#)

**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** • [Juniper Networks Devices Processing Overview on page 3](#)

## ethernet-switching

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
ethernet-switching {
  block-non-ip-all;
  bpdu-vlan-flooding;
  bypass-non-ip-unicast;
  no-packet-flooding {
    no-trace-route;
  }
}
```

**Hierarchy Level** [edit security flow]

**Release Information** Statement introduced in Junos OS Release 9.5.

**Description** Changes default Layer 2 forwarding behavior.

- Options**
- **block-non-ip-all**—Block all Layer 2 non-IP and non-ARP traffic, including multicast and broadcast traffic.
  - **bypass-non-ip-unicast**—Allow all Layer 2 non-IP traffic to pass through the device.
  - **no-packet-flooding**—Stop IP flooding and send ARP or ICMP requests to discover the destination MAC address for a unicast packet.



**NOTE:** On all SRX Series devices in transparent mode, packet flooding is enabled by default. If you have manually disabled packet flooding with the `set security flow ethernet-switching no-packet-flooding` command, then multicast packets such as OSPFv3 hello packets are dropped.

- **no-trace-route**—Do not send ICMP requests to discover the destination MAC address for a unicast packet. Only ARP requests are sent. This option only allows the device to discover the destination MAC address for a unicast packet if the destination IP address is in the same subnetwork as the ingress IP address.



**NOTE:** The **block-non-ip-all** and **bypass-non-ip-unicast** options cannot be configured at the same time.

**Required Privilege Level**

security—To view this in the configuration.

security-control—To add this to the configuration.



- Related Documentation**
- [Juniper Networks Devices Processing Overview on page 3](#)

## destination-header

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 252](#)
  - [hop-by-hop-header on page 247](#)

## destination-port (Security Forwarding Options)

---

**Supported Platforms**    [SRX Series, vSRX](#)

**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 18 on page 227](#).

Table 18: 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
kpasswd	761
krb-prop	754
krbupdate	760

Table 18: Ports Supported by Services Interfaces (*continued*)

Port Name	Corresponding Port Number
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
radius	1812
rip	520
rkinit	2108
smtp	25
snmp	161
snmp-trap	162

Table 18: Ports Supported by Services Interfaces (*continued*)

Port Name	Corresponding Port Number
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
xmcp	177

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

**Related Documentation**

- [mirror-filter \(Security Forwarding Options\) on page 258](#)
- [show security forwarding-options mirror-filter on page 573](#)

## destination-prefix (Security Forwarding Options)

---

Supported Platforms	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
Syntax	destination-prefix <i>destination-prefix</i> ;
Hierarchy Level	[edit security forwarding-options mirror-filter <i>filter-name</i> ]
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	<ul style="list-style-type: none"><li>• <a href="#">mirror-filter (Security Forwarding Options) on page 258</a></li><li>• <a href="#">show security forwarding-options mirror-filter on page 573</a></li></ul>

## early-ageout

---

Supported Platforms	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
Syntax	early-ageout <i>seconds</i> ;
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	<b>seconds</b> —Amount of time that elapses before the device aggressively ages out a session. <b>Range:</b> 1 through 65,535 seconds <b>Default:</b> 20 seconds
Required Privilege Level	security—To view this in the configuration. security-control—To add this to the configuration.
Related Documentation	<ul style="list-style-type: none"><li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li></ul>

## error

**Supported Platforms** SRX5400, SRX5600, SRX5800

**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 RGI 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 19 on page 232 shows the range of values for each error level.

**Table 19: 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.

<b>Required Privilege Level</b>	<b>interface</b> —To view this in the configuration.
	<b>interface-control</b> —To add this to the configuration.

<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">fpc error on page 241</a></li></ul>
------------------------------	---

## fin-invalidate-session

---

**Supported Platforms** [SRX Series, vSRX](#)

**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(nowledge) 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 20 on page 235](#) 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 20: Terminating a Session with a 4-Way Handshake

Step	Client	Server
1.	FIN	
2.		ACK
3.		FIN <ul style="list-style-type: none"><li>• Sets session timer to 150 seconds.</li></ul>
4.	ACK <ul style="list-style-type: none"><li>• Sets session timer to 2 seconds.</li></ul>	

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 <ul style="list-style-type: none"><li>• sets session timer to 150 seconds</li></ul>
3.	ACK <ul style="list-style-type: none"><li>• sets session timer to 2 seconds</li></ul>	

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

**Related Documentation**

- [Juniper Networks Devices Processing Overview on page 3](#)

## flow (Security Flow)

---

Supported Platforms [SRX Series, vSRX](#)

```
Syntax flow {
    aging {
        early-ageout seconds;
        high-watermark percent;
        low-watermark percent;
    }
    allow-dns-reply;
    ethernet-switching {
        block-non-ip-all;
        bpdv-vlan-flooding;
        bypass-non-ip-unicast;
        no-packet-flooding {
            no-trace-route;
        }
    }
    force-ip-reassembly;
    ipsec-performance-acceleration;
    load distribution {
        session-affinity ipsec;
    }
    packet-log {
        enable;
        throttle-interval;
        packet-filter <filter-name>;
    }
    pending-sess-queue-length (high | moderate | normal);
    route-change-timeout seconds;
    syn-flood-protection-mode (syn-cookie | syn-proxy);
    tcp-mss {
        all-tcp mss value;
        gre-in {
            mss value;
        }
        gre-out {
            mss value;
        }
        ipsec-vpn {
            mss value;
        }
    }
    tcp-session {
        fin-invalidate-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);
    }
}
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 {
        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;
}
}

```

**Hierarchy Level** [edit security]

**Release Information** Statement modified in Junos OS Release 9.5.

**Description** Determine how the device manages packet flow. The device can regulate packet flow in the following ways:

**Options** The remaining statements are explained separately. See [CLI Explorer](#).

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

**Related Documentation**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [Understanding Session Characteristics for SRX Series Services Gateways on page 33](#)
- [Understanding Flow in Logical Systems for SRX Series Devices](#)

## force-ip-reassembly

---

**Supported Platforms**   [SRX Series, vSRX](#)

**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**   • [Juniper Networks Devices Processing Overview on page 3](#)

## forwarding-process

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
forwarding-process {
  application-services {
    enable-gtpu-distribution;
    maximize-alg-sessions;
    maximize-idp-sessions {
      weight (firewall | idp);
    }
    packet-ordering-mode {
      (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**     security—To view this in the configuration.  
**Level**     security-control—To add this to the configuration.

**Related Documentation**

- [application-services \(Security Forwarding Process\) on page 222](#)
- [Juniper Networks Devices Processing Overview on page 3](#)



## fpc error

**Supported Platforms** SRX Series, vSRX

**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.

<b>Required Privilege Level</b>	routing—To view this in the configuration. routing-control—To add this to the configuration.
---------------------------------	---

<b>Related Documentation</b>	• <a href="#">error on page 231</a>
------------------------------	-------------------------------------

## fru-poweron-sequence

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## gre-in

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## gre-out

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## high-watermark

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## hop-by-hop-header

**Supported Platforms** [SRX Series, vSRX](#)

**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 Multicast 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 on page 252](#)
- [destination-header on page 225](#)

## icmpv6-malformed

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 252](#)



---

## idle-timeout (System Services)

---

**Supported Platforms**   [SRX Series](#)

**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:** 1440  
              **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

---

<b>Supported Platforms</b>	<a href="#">SRX Series, vSRX</a>
<b>Syntax</b>	inline-tap;
<b>Hierarchy Level</b>	[edit security forwarding-process application-services maximize-idp-sessions]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.2.
<b>Description</b>	<p>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.</p> <p>This statement is supported in SRX1500, SRX 5800, SRX 5600, and SRX 5400 devices and vSRX.</p>
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li></ul>

## interface-in (Security Forwarding Options)

---

<b>Supported Platforms</b>	<a href="#">SRX Series, vSRX</a>
<b>Syntax</b>	interface-in <i>interface-name</i> ;
<b>Hierarchy Level</b>	[edit security forwarding-options mirror-filter <i>filter-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.1X46-D10.
<b>Description</b>	Specify the incoming logical interface to be matched for mirroring.
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">mirror-filter (Security Forwarding Options) on page 258</a></li><li>• <a href="#">show security forwarding-options mirror-filter on page 573</a></li></ul>

## interface-out (Security Forwarding Options)

---

<b>Supported Platforms</b>	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
<b>Syntax</b>	interface-out <i>interface-name</i> ;
<b>Hierarchy Level</b>	[edit security forwarding-options mirror-filter <i>filter-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.1X46-D10.
<b>Description</b>	Specify the outgoing logical interface to be matched for mirroring.
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">mirror-filter (Security Forwarding Options) on page 258</a></li> <li>• <a href="#">show security forwarding-options mirror-filter on page 573</a></li> </ul>

## ipv4-template (Services)

---

<b>Supported Platforms</b>	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
<b>Syntax</b>	ipv4-template;
<b>Hierarchy Level</b>	[edit services flow-monitoring version9 template <i>template-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.4.
<b>Description</b>	Specify that the flow monitoring version 9 template is used only for IPv4 records.
<b>Required Privilege Level</b>	services—To view this in the configuration. services-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li> <li>• <a href="#">Understanding Interfaces</a></li> </ul>

## ipv6-extension-header

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 247](#)
- [destination-header on page 225](#)

## ipv6-extension-header-limit

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 252](#)

## ipv6-malformed-header

---

<b>Supported Platforms</b>	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
<b>Syntax</b>	ipv6-malformed-header;
<b>Hierarchy Level</b>	[edit security screen ids-option <i>screen-name</i> ip]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.1X46-D10.
<b>Description</b>	Enable the IPv6 malformed header intrusion detection service (IDS) option.
<b>Options</b>	This has no options.
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Understanding IPv6 Support for Screens</a></li><li>• <a href="#">ipv6-extension-header on page 252</a></li></ul>

## ipv6-template (Services)

---

<b>Supported Platforms</b>	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
<b>Syntax</b>	ipv6-template;
<b>Hierarchy Level</b>	[edit services flow-monitoring version9 template <i>template-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.1X45-D10.
<b>Description</b>	Specify that the flow monitoring version 9 template is used only for IPv6 records.
<b>Required Privilege Level</b>	services—To view this in the configuration. services-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li><li>• <a href="#">Understanding Interfaces</a></li></ul>

## low-latency

---

**Supported Platforms** [SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, SRX5800](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [Example: Configuring Low Latency on page 117](#)

## low-watermark

---

**Supported Platforms** [SRX Series, vSRX](#)

**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** security—To view this in the configuration.

**Level** security-control—To add this to the configuration.

**Related Documentation**

- [Juniper Networks Devices Processing Overview on page 3](#)



## maximize-idp-sessions

**Supported Platforms** [SRX Series](#)

**Syntax** `maximize-idp-sessions {  
weight (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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## mirror-filter (Security Forwarding Options)

**Supported Platforms** SRX Series, vSRX

**Syntax**

```
mirror-filter filter-name {
    destination-port port-number;
    destination-prefix destination-prefix;
    interface-in interface-name;
    interface-out interface-name;
    output {
        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.

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

**Related Documentation**

- [show security forwarding-options mirror-filter on page 573](#)
- [clear security forward-options mirror filter on page 326](#)

## mode (Security Forwarding Options)

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 21 on page 261](#) 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 21: 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** • [Juniper Networks Devices Processing Overview on page 3](#)

## no-sequence-check

**Supported Platforms** [SRX Series, vSRX](#)

**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** • [Juniper Networks Devices Processing Overview on page 3](#)

## no-tcp-reset

---

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax**

```
no-tcp-reset {  
    drop-all-tcp;  
    drop-tcp-with-sync-only;  
}
```

**Hierarchy Level** [edit system internet-options]

**Release Information** Statement introduced in Junos OS Release 11.1.

**Description** Do not send reset RST TCP packets for packets sent to non-listening ports.

- Options**
- **drop-all-tcp**—Drop all TCP packets.
  - **drop-tcp-with-sync-only**— Drop TCP packets with a SYN bit.

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

- Related Documentation**
- *Administration Guide for Security Devices*
  - [Juniper Networks Devices Processing Overview on page 3](#)

---

## output (Security Forwarding Options)

---

**Supported Platforms** [SRX Series, vSRX](#)

**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.

**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 \(Security Forwarding Options\) on page 258](#)
- [show security forwarding-options mirror-filter on page 573](#)

## packet-filter

---

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax** `packet-filter packet-filter-name {  
 action-profile (profile-name | default);  
 destination-port (port-range | protocol-name);  
 destination-prefix destination-prefix;  
 interface logical-interface-name;  
 protocol (protocol-number | protocol-name);  
 source-port (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. Support for IPv6 filter for the **interface** option added in Junos OS Release 10.4. This statement is supported on SRX4100, SRX4200, SRX5800, SRX5600 devices and vSRX.

**Description** Set packet filter for taking the datapath-debug action. A maximum of four filters are supported at the same time.

This statement is supported on SRX4100, SRX 5800, SRX 5600, and SRX 5400 devices and vSRX.

- Options**
- **action-profile** (*profile-name* | default)—Identify the action profile to use. You can specify the name of the action profile to use or select default action profile.
  - **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)

**Supported Platforms** [SRX Series](#)

**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)

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## pending-sess-queue-length

---

<b>Supported Platforms</b>	SRX Series, vSRX
<b>Syntax</b>	pending-sess-queue-length (high   moderate   normal);
<b>Hierarchy Level</b>	[edit security flow]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.4.
<b>Description</b>	Configure the maximum queued length per pending session.
<b>Options</b>	<ul style="list-style-type: none"> <li>• <b>high</b>— Allow the maximum number of queued sessions.</li> <li>• <b>moderate</b>—Allow more queued sessions than the normal number.</li> <li>• <b>normal</b>—Allow the normal number of queued session.</li> </ul>
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li> </ul>

## per-session-statistics

---

<b>Supported Platforms</b>	SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, SRX5800
<b>Syntax</b>	per-session-statistics;
<b>Hierarchy Level</b>	[edit chassis fpc <i>name name</i> pic <i>name name</i> services-offload]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.4.
<b>Description</b>	<p>Keeps per session statistics in NPC monitoring.</p> <p>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.</p>
<b>Required Privilege Level</b>	security—To view this in the configuration. security-control—To add this to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li> <li>• <a href="#">services-offload on page 276</a></li> </ul>

## preserve-incoming-fragment-size

---

**Supported Platforms** [SRX Series, vSRX](#)

**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.

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 on page 52](#)

## propagate-settings

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## protocol (Security Forwarding Options)

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 \(Security Forwarding Options\) on page 258](#)
- [show security forwarding-options mirror-filter on page 573](#)

---

## resource-manager

---

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax**

```
resource-manager {  
  traceoptions {  
    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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## route-change-timeout

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)



---

## rst-invalidate-session

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## rst-sequence-check

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## sampling

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## services-offload

---

**Supported Platforms** [SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, SRX5800](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## session (System Services)

**Supported Platforms** [SRX Series](#)

**Syntax**

```
session {
    idle-timeout 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)

---

**Supported Platforms**   [SRX Series](#)

**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*

## np-cache (Flexible PIC Concentrator)

**Supported Platforms** [SRX Series, vSRX](#)

**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 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 on page 107](#)
- [Example: Configuring SRX5K-MPC3-100G10G \(IOC3\) and SRX5K-MPC3-40G10G \(IOC3\) on an SRX5000 Line Device to Support Express Path on page 109](#)

## source-port (Security Forwarding Options)

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 22 on page 281](#).



Table 22: 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
kpasswd	761
krb-prop	754
krbupdate	760

Table 22: Ports Supported by Services Interfaces (*continued*)

Port Name	Corresponding Port Number
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
nnntp	119
ntalk	518
ntp	123
pop3	110
pptp	1723
printer	515
radacct	1813
radius	1812
rip	520
rkinit	2108
smtp	25
snmp	161
snmp-trap	162

Table 22: Ports Supported by Services Interfaces (*continued*)

Port Name	Corresponding Port Number
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 \(Security Forwarding Options\) on page 258](#)
- [show security forwarding-options mirror-filter on page 573](#)

## source-prefix (Security Forwarding Options)

---

Supported Platforms	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
Syntax	<code>source-prefix <i>source-prefix</i>;</code>
Hierarchy Level	<code>[edit security forwarding-options mirror-filter <i>filter-name</i>]</code>
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	<ul style="list-style-type: none"><li>• <a href="#">mirror-filter (Security Forwarding Options) on page 258</a></li><li>• <a href="#">show security forwarding-options mirror-filter on page 573</a></li></ul>

## syn-flood-protection-mode

---

Supported Platforms	<a href="#">SRX Series</a> , <a href="#">vSRX</a>
Syntax	<code>syn-flood-protection-mode (syn-cookie   syn-proxy);</code>
Hierarchy Level	<code>[edit security flow]</code>
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 <b>syn-flood attack-threshold value</b> is exceeded.
Options	<ul style="list-style-type: none"><li>• <b>syn-cookie</b>—Uses a cryptographic hash to generate a unique Initial Sequence Number (ISN). This is enabled by default.</li><li>• <b>syn-proxy</b>—Uses a proxy to handle the SYN attack.</li></ul>
Required Privilege Level	security—To view this in the configuration. security-control—To add this to the configuration.

## tcp-initial-timeout

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## tcp-mss (Security Flow)

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 220](#)
- [gre-in on page 244](#)

- [gre-out on page 245](#)
- *ipsec-vpn (Security Flow)*
- [Juniper Networks Devices Processing Overview on page 3](#)

## tcp-session

**Supported Platforms** [SRX Series, vSRX](#)

**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** • [Juniper Networks Devices Processing Overview on page 3](#)

## time-wait-state

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)



## traceoptions (Security)

**Supported Platforms** SRX Series, vSRX

**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.

<b>Required Privilege</b>	trace—To view this in the configuration.
<b>Level</b>	trace-control—To add this to the configuration.

<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li></ul>
------------------------------	--

## traceoptions (Security Flow)

**Supported Platforms** SRX Series, vSRX

**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]

**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. The was updated in Junos OS Release 15.1X49-D70 with the addition of the conn-tag filter parameter.

**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.

<b>Required Privilege</b>	trace—To view this in the configuration.
<b>Level</b>	trace-control—To add this to the configuration.

<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Juniper Networks Devices Processing Overview on page 3</a></li></ul>
------------------------------	--

## transport (Security Log)

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax**

```
transport {
  protocol (udp | tcp | tls);
  tls-profile tls-profile-name;
  tcp-connections tcp-connections;
}
```

**Hierarchy Level** [edit security log]

**Release Information** Statement introduced in Junos OS Release 12.1X46-D25.

**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)

**Supported Platforms** [SRX Series, vSRX](#)

**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 23: 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
<b>equal</b>	1,000,000	1,000,000	8.5 Gbps	2 Gbps
<b>firewall</b>	1,000,000	1,000,000	10 Gbps	2.4 Gbps
<b>idp</b>	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**

- [Juniper Networks Devices Processing Overview on page 3](#)



## CHAPTER 14

# Operational Commands

- clear firewall
- clear monitor security flow filter
- clear security flow ip-action
- clear security flow session all
- clear security flow session application
- clear security flow session destination-port
- clear security flow session family
- clear security flow session interface
- clear security flow session protocol
- clear security flow session resource-manager
- clear security flow session services-offload
- clear security flow session session-identifier
- clear security flow session source-port
- clear security flow session source-prefix
- clear security forward-options mirror filter
- monitor security flow file
- monitor security flow filter
- monitor security flow start
- monitor security flow stop
- show chassis environment (Security)
- show chassis fpc (View)
- show chassis fpc errors
- show chassis hardware (View)
- show chassis pic (Security)
- show chassis power
- show chassis power sequence
- show firewall (View)
- show interfaces (View Aggregated Ethernet)

- `show interfaces (SRX Series)`
- `show interfaces diagnostics optics`
- `show interfaces flow-statistics`
- `show interfaces swfabx`
- `show monitor security flow`
- `show security flow cp-session`
- `show security flow cp-session destination-port`
- `show security flow cp-session destination-prefix`
- `show security flow cp-session family`
- `show security flow cp-session protocol`
- `show security flow cp-session source-port`
- `show security flow cp-session source-prefix`
- `show security flow gate`
- `show security flow ip-action`
- `show security flow gate brief node`
- `show security flow gate destination-port`
- `show security flow gate destination-prefix`
- `show security flow gate protocol`
- `show security flow gate summary node`
- `show security flow session`
- `show security flow session brief node`
- `show security flow session destination-port`
- `show security flow session destination-prefix`
- `show security flow session extensive node`
- `show security flow session family`
- `show security flow session interface`
- `show security flow session nat`
- `show security flow session policy-id`
- `show security flow session protocol`
- `show security flow session resource-manager`
- `show security flow session services-offload`
- `show security flow session session-identifier`
- `show security flow session source-port`
- `show security flow session source-prefix`
- `show security flow session summary family`
- `show security flow session summary node`
- `show security flow session summary services-offload`

- `show security flow session tunnel`
- `show security flow statistics`
- `show security flow status`
- `show security forwarding-options mirror-filter`
- `show security monitoring`
- `show security policies`
- `show security policies hit-count`
- `show security resource-manager group active`
- `show security resource-manager resource active`
- `show security resource-manager settings`
- `show security resource-manager summary`
- `show security screen ids-option`
- `show security screen statistics`
- `show security softwires`
- `show security zones`
- `show security zones type`

## clear firewall

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 \(View\) on page 364](#)

**List of Sample Output** [clear firewall all on page 300](#)

**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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 161](#)
- [monitor security flow start on page 331](#)
- [monitor security flow filter on page 329](#)

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

## Sample Output

## clear security flow ip-action

---

**Supported Platforms** [SRX Series](#), [vSRX](#)

**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 on page 448](#)

**List of Sample Output**

[clear security flow ip-action all on page 303](#)  
[clear security flow ip-action destination-prefix on page 303](#)  
[clear security flow ip-action family inet on page 303](#)  
[clear security flow ip-action protocol udp on page 303](#)

**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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 476](#)

**List of Sample Output** [clear security flow session all on page 304](#)  
[clear security flow session all node 0 on page 305](#)

**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

---

**Supported Platforms** [SRX Series](#), [vSRX](#)

**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.

<b>Required Privilege Level</b>	clear
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>show security flow session application</i></li> </ul>
<b>List of Sample Output</b>	<a href="#">clear security flow session application dns on page 307</a> <a href="#">clear security flow session application dns node 0 on page 307</a>
<b>Output Fields</b>	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 destination-port

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 487](#)

**List of Sample Output** [clear security flow session destination-port 1 on page 308](#)  
[clear security flow session destination-port 1 node 0 on page 309](#)

**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 family

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 501](#)

**List of Sample Output** [clear security flow session family inet on page 310](#)  
[clear security flow session family inet6 on page 310](#)

**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 interface

Supported Platforms	SRX Series, vSRX
Syntax	clear security flow session interface <i>interface-name</i> <node ( <i>node-id</i>   all   local   primary)>
Release Information	Command introduced in Junos OS Release 8.5; <b>node</b> options added in Junos OS Release 9.0.
Description	Clear sessions that use the specified interface.
Options	<ul style="list-style-type: none"> <li>• <b><i>interface-name</i></b> —Name of a specific incoming or outgoing interface.</li> <li>• <b>node</b>—(Optional) For chassis cluster configurations, clear security sessions on the interface on a specific node (device) in the cluster. <ul style="list-style-type: none"> <li>• <b><i>node-id</i></b> —Identification number of the node. It can be 0 or 1.</li> <li>• <b>all</b> —Clear all nodes.</li> <li>• <b>local</b> —Clear the local node.</li> <li>• <b>primary</b>—Clear the primary node.</li> </ul> </li> </ul>
Required Privilege Level	clear
Related Documentation	<ul style="list-style-type: none"> <li>• <a href="#">show security flow session interface on page 506</a></li> </ul>
List of Sample Output	<a href="#">clear security flow session interface ge-0/0/0.0 on page 311</a> <a href="#">clear security flow session interface ge/0/0.0 node 0 on page 312</a>
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 session protocol

**Supported Platforms** SRX Series, vSRX

**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 on page 516](#)

**List of Sample Output**    [clear security flow session protocol pim on page 314](#)  
                                  [clear security flow session protocol 0 on page 314](#)

**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

Supported Platforms	SRX Series, vSRX
Syntax	clear security flow session resource-manager <node ( <i>node-id</i>   all   local   primary )>
Release Information	Command introduced in Junos OS Release 8.5; <b>node</b> options added in Junos OS Release 9.0.
Description	Clear resource-manager sessions.
Options	<ul style="list-style-type: none"> <li><b>node</b>—(Optional) For chassis cluster configurations, clear the resource manager sessions on a specific node (device) in the cluster. <ul style="list-style-type: none"> <li><b>node-id</b>—Identification number of the node. It can be 0 or 1.</li> <li><b>all</b>—Clear all nodes.</li> <li><b>local</b>—Clear the local node.</li> <li><b>primary</b>—Clear the primary node.</li> </ul> </li> </ul>
Required Privilege Level	clear
Related Documentation	<ul style="list-style-type: none"> <li><a href="#">show security flow session resource-manager on page 521</a></li> </ul>
List of Sample Output	<a href="#">clear security flow session resource-manager on page 315</a> <a href="#">clear security flow session resource-manager node 0 on page 315</a>
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

**Supported Platforms** [SRX Series](#), [vSRX](#)

**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 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 on page 525](#)

**List of Sample Output**

[clear security flow session services-offload on page 318](#)  
[clear security flow session services-offload application on page 318](#)  
[clear security flow session services-offload destination-port on page 318](#)  
[clear security flow session services-offload destination-prefix on page 319](#)  
[clear security flow session services-offload family on page 319](#)  
[clear security flow session services-offload interface on page 319](#)  
[clear security flow session services-offload logical-system on page 319](#)  
[clear security flow session services-offload protocol on page 319](#)  
[clear security flow session services-offload root-logical-system on page 319](#)  
[clear security flow session services-offload source-port on page 319](#)  
[clear security flow session services-offload source-prefix on page 319](#)

**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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 530](#)

**List of Sample Output** [clear security flow session session-identifier 1 on page 320](#)  
[clear security flow session session-identifier 1 node 0 on page 320](#)

**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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 535](#)

**List of Sample Output** [clear security flow session source-port 1 on page 322](#)  
[clear security flow session source-port 1 node 0 on page 323](#)

**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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 539](#)

**List of Sample Output** [clear security flow session source-prefix 10.0.0.1 on page 324](#)  
[clear security flow session source-prefix 10::10 on page 325](#)  
[clear security flow session source-prefix 10.0.0.1 node 0 on page 325](#)

**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 forward-options mirror filter

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 \(Security Forwarding Options\) on page 258](#)
- [show security forwarding-options mirror-filter on page 573](#)

## monitor security flow file

**Supported Platforms** SRX Series, vSRX

**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 on page 161](#)
  - [monitor security flow filter on page 329](#)
  - [monitor security flow start on page 331](#)
  - [show monitor security flow on page 421](#)
- Output Fields**    This command produces no output.



## monitor security flow filter

**Supported Platforms** SRX Series, vSRX

**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 on page 161](#)
- [monitor security flow file on page 327](#)
- [monitor security flow start on page 331](#)
- [monitor security flow stop on page 332](#)

---

## monitor security flow start

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 161](#)
- [show monitor security flow on page 421](#)
- [monitor security flow filter on page 329](#)
- [monitor security flow stop on page 332](#)

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

## monitor security flow stop

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 161](#)
- [monitor security flow start on page 331](#)

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

## show chassis environment (Security)

**Supported Platforms** [SRX Series, vSRX](#)

**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).

**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 \(View\) on page 347](#)

**List of Sample Output** [show chassis environment on page 334](#)

**Output Fields** [Table 24 on page 333](#) lists the output fields for the **show chassis environment** command. Output fields are listed in the approximate order in which they appear.

**Table 24: 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: <b>OK</b> , <b>Testing</b> (during initial power-on), <b>Failed</b> , or <b>Absent</b> .

## 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		



## show chassis fpc (View)

Supported Platforms [SRX Series](#)

**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 25 on page 337](#) summarizes the SRX5K-MPC3-40G10G (IOC3) PICs selected for various configuration scenarios.

**Table 25: 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
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

**Table 25: SRX5K-MPC3-40G10G (IOC3) PIC Selection Summary** (*continued*)

CLI Configuration	PIC Selection
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 340](#)  
[show chassis fpc \(SRX5600 and SRX5800 devices\) on page 340](#)  
[show chassis fpc \(SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G \(IOC3\) or SRX5K-MPC3-40G10G \(IOC3\) on page 340](#)  
[show chassis fpc detail 2 on page 341](#)  
[show chassis fpc pic-status \(SRX5600 and SRX5800 devices\) on page 341](#)  
[show chassis fpc pic-status \(SRX5600 and SRX5800 devices with SPC2\) on page 341](#)  
[show chassis fpc pic-status \(SRX5600 and SRX5800 devices with SRX5K-MPC\) on page 342](#)  
[show chassis fpc pic-status \(SRX5600 and SRX5800 devices when Express Path \[formerly known as services offloading\] is configured\) on page 342](#)  
[show chassis fpc pic-status \(with 20-Gigabit Ethernet MIC with SFP\) on page 343](#)  
[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 343](#)  
[show chassis fpc pic-status for HA \(SRX5600 and SRX5800 devices\) on page 343](#)  
[show chassis fpc pic-status for HA \(SRX5400, SRX5600, and SRX5800 devices with SRX5K-MPC3-100G10G \(IOC3\) or SRX5K-MPC3-40G10G \(IOC3\) on page 344](#)

**Output Fields** [Table 26 on page 339](#) lists the output fields for the **show chassis fpc** command. Output fields are listed in the approximate order in which they appear.

**Table 26: 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: <ul style="list-style-type: none"> <li>• <b>Dead</b>—Held in reset because of errors.</li> <li>• <b>Diag</b>—Slot is being ignored while the device is running diagnostics.</li> <li>• <b>Dormant</b>—Held in reset.</li> <li>• <b>Empty</b>—No FPC is present.</li> <li>• <b>Online</b>—FPC is online and running.</li> <li>• <b>Present</b>—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 <b>Hardware Not Supported</b> or <b>Hardware Not In Right Slot</b>. FPC is coming up but not yet online.</li> <li>• <b>Probed</b>—Probe is complete; awaiting restart of the Packet Forwarding Engine (PFE).</li> <li>• <b>Probe-wait</b>—Waiting to be probed.</li> </ul>
Temp (C) or Temperature	Temperature of the air passing by the FPC, in degrees Celsius or in both Celsius and Fahrenheit.

Table 26: show chassis fpc Output Fields (*continued*)

Field Name	Field Description
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 (%)
          (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 (%)
          (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
```

Slot	State	Temp (C)	CPU Utilization (%) Total	CPU Utilization (%) Interrupt	CPU Utilization (%) 1min	CPU Utilization (%) 5min	CPU Utilization (%) 15min	Memory DRAM (MB)
0	Online	36	20	Heap 0 Buffer 20	19	19	1024	
1	Online	35	8	4 0	26 8	8	8	2048
2	Online	40	21	12 0	14 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

**Supported Platforms** [SRX5400, SRX5600, SRX5800](#)

**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 on page 241](#)

**List of Sample Output** [show chassis fpc errors on page 346](#)

**Output Fields** [Table 27 on page 345](#) lists the output fields for the **show chassis fpc errors** command. Output fields are listed in the approximate order in which they appear.

**Table 27: show chassis fpc errors Output Fields**

Field Name	Field Description
FPC	The FPC number.
Level	The severity of the error. It can be configured as follows: <ul style="list-style-type: none"> <li>• <b>fatal</b>—Fatal error on the FPC.</li> <li>• <b>major</b>—Major error on the FPC.</li> <li>• <b>minor</b>—Minor error on the FPC.</li> </ul>
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.

Table 27: show chassis fpc errors Output Fields (*continued*)

Field Name	Field Description
<b>Action</b>	<p>The detection and recovery actions that are triggered when the threshold value is exceeded.</p> <ul style="list-style-type: none"> <li>Restart the FPC.</li> <li>Get the current state of the FPC and raise an alarm.</li> <li>Write a log for the event.</li> </ul>

## 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)

**Supported Platforms** [SRX Series](#)

**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**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [Interface Naming Conventions](#)

**Output Fields** [Table 28 on page 347](#) lists the output fields for the **show chassis hardware** command. Output fields are listed in the approximate order in which they appear.

**Table 28: 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.

Table 28: show chassis hardware Output Fields (*continued*)

Field Name	Field Description
<b>Serial Number</b>	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.
<b>Assb ID or Assembly ID</b>	Identification number that describes the FRU hardware.
<b>FRU model number</b>	Model number of FRU hardware component.
<b>CLEI code</b>	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.
<b>EEPROM Version</b>	ID EEPROM version used by hardware component: 0x01 (version 1) or 0x02 (version 2).

Table 28: show chassis hardware Output Fields (*continued*)

Field Name	Field Description
Description	<p>Brief description of the hardware item:</p> <ul style="list-style-type: none"> <li>Type of power supply.</li> <li>Switch Control Board (SCB)</li> </ul> <p>Starting with Junos OS Release 12.1X47-D15 and Junos OS Release 17.3R1, the SRX5K-SCBE (SCB2) is introduced.</p> <ul style="list-style-type: none"> <li>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.</li> <li>With an SCB2, a third SCB is supported. If a third SCB is plugged in, it provides intra-chassis fabric redundancy.</li> <li>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.</li> <li>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.</li> <li>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.</li> </ul> <p>Starting with Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, the SRX5K-SCB3 (SCB3) with enhanced midplane is introduced.</p> <ul style="list-style-type: none"> <li>All existing SCB software that is supported by SCB2 is supported on SCB3.</li> <li>SRX5K-RE-1800X4 (RE2). Mixed Routing Engine use is not supported.</li> <li>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.</li> <li>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.</li> <li>SCB3 supports up to 400 Gbps per slot with old midplanes and up to 500 Gbps per slot with new midplanes.</li> <li>SCB3 supports fabric intra-chassis redundancy.</li> <li>SCB3 supports the same chassis cluster function as the SRX5K-SCB (SCB1) and the SRX5K-SCBE (SCB2), except for in-service software upgrade (ISSU) and in-service hardware upgrade (ISHU).</li> <li>SCB3 has a second external Ethernet port.</li> <li>Fabric bandwidth increasing mode is not supported.</li> </ul>

Table 28: show chassis hardware Output Fields (*continued*)

Field Name	Field Description
	<ul style="list-style-type: none"> <li>Type of Flexible PIC Concentrator (FPC), Physical Interface Card (PIC), Modular Interface Cards (MICs), and PIMs.</li> <li>IOCs           <p>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.</p> <ul style="list-style-type: none"> <li>IOC3 has two types of IOC3 MPCs, which have different built-in MICs: the 24x10GE + 6x40GE MPC and the 2x100GE + 4x10GE MPC.</li> <li>IOC3 supports SCB3 and SRX5000 line backplane and enhanced backplane.</li> <li>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.</li> <li>IOC3 interoperates with SCB2 and SCB3.</li> <li>IOC3 interoperates with the SRX5K-SPC-4-15-320 (SPC2) and the SRX5K-MPC (IOC2).</li> <li>The maximum power consumption for one IOC3 is 645W. An enhanced power module must be used.</li> <li>The IOC3 does not support the following command to set a PIC to go offline or online:  <b>request chassis pic fpc-slot &lt;fpc-slot&gt; pic-slot &lt;pic-slot&gt; &lt;offline   online&gt; .</b> </li> <li>IOC3 supports 240 Gbps of throughput with the enhanced SRX5000 line backplane.</li> <li>Chassis cluster functions the same as for the SRX5000 line IOC2.</li> <li>IOC3 supports intra-chassis and inter-chassis fabric redundancy mode.</li> <li>IOC3 supports ISSU and ISHU in chassis cluster mode.</li> <li>IOC3 supports intra-FPC and Inter-FPC Express Path (previously known as <i>services offloading</i>) with IPv4.</li> <li>NAT of IPv4 and IPv6 in normal mode and IPv4 for Express Path mode.</li> <li>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 <b>set chassis fpc &lt;slot&gt; pic &lt;pic&gt; power off</b> command to choose the PICs you want to power on.</li> </ul> <p><b>NOTE:</b> Fabric bandwidth increasing mode is not supported on IOC3.</p> </li> <li>SRX Clustering Module (SCM)</li> <li>Fan tray</li> <li>For hosts, the Routing Engine type.           <ul style="list-style-type: none"> <li>Starting with Junos OS Release 12.1X47-D15 and Junos OS Release 17.3R1, the SRX5K-RE-1800X4 (RE2) Routing Engine is introduced.</li> <li>The RE2 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.</li> </ul> <p><b>NOTE:</b> The RE2 provides significantly better performance than the previously used Routing Engine, even with a single core.</p> </li> </ul>

## show chassis hardware

### show chassis hardware

```

user@host> show chassis hardware
Hardware inventory:
Item             Version  Part number  Serial number  Description
Chassis                               CM0715AK0021  SRX1500
Midplane                               ACMA4255      SRX1500
CB 0                               711-053838    ACMA7529      CPU Board SRX700E
Routing Engine 0 BUILTIN      BUILTIN      SRX Routing Engine
FPC 0                               711-053832    ACMA3311      FEB
  PIC 0                               BUILTIN      BUILTIN      12x1G-T-4x1G-SFP-4x10G
    Xcvr 12                           REV 01      740-014132    61521013      SFP-T
    Xcvr 13                           REV 02      740-013111    A281604       SFP-T
    Xcvr 14                           REV 02      740-011613    NRN30NV       SFP-SX
    Xcvr 15                           REV 02      740-011613    NRN2PWV       SFP-SX
    Xcvr 16                           REV 01      740-021308    AJA17B5       SFP+-10G-SR
    Xcvr 17                           REV 01      740-021308    MSP056B       SFP+-10G-SR
    Xcvr 18                           REV 01      740-031980    AS920WJ       SFP+-10G-SR
    Xcvr 19                           REV 01      740-031980    AS92W5N       SFP+-10G-SR
Power Supply 0   REV 01      740-055217    1EDP42500JZ   PS 400W 90-264V AC in
Fan Tray 0                                             SRX1500 0, Front to Back
  Airflow - AFO
Fan Tray 1                                             SRX1500 1, Front to Back
  Airflow - AFO
Fan Tray 2                                             SRX1500 2, Front to Back
  Airflow - AFO
Fan Tray 3                                             SRX1500 3, Front to Back
  Airflow - AFO

```

### 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                               ACAX3849      SRX 5800 Backplane
FPM Board                               CAAX7297      Front Panel Display
PDM                               Rev 03      740-013110    QCS170250DU    Power Distribution Module
PEM 0                               Rev 03      740-034724    QCS17020203F   PS 4.1kW; 200-240V AC in
PEM 1                               Rev 03      740-034724    QCS17020203C   PS 4.1kW; 200-240V AC in
PEM 2                               Rev 04      740-034724    QCS17100200A   PS 4.1kW; 200-240V AC in
PEM 3                               Rev 03      740-034724    QCS17080200M   PS 4.1kW; 200-240V AC in
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	AMBOHG3	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&gt; 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	MUCOMF2	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	RMPD 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	AN40MVB	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
PEM 1	Rev 03	740-034701	QCS13090904T	PS 1.4-2.6kW; 90-264V A
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	ALNON6C	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 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	COUIBCBAA	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)

**Supported Platforms** [SRX Series](#)

**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 Feature Guide for Security Devices](#)
- [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show chassis pic fpc-slot pic-slot on page 359](#)

**Output Fields** [Table 29 on page 358](#) lists the output fields for the **show chassis pic** command. Output fields are listed in the approximate order in which they appear.

**Table 29: show chassis pic Output Fields**

Field Name	Field Description
<b>Type</b>	PIC type.
<b>State</b>	Status of the PIC. State is displayed only when a PIC is in the slot. <ul style="list-style-type: none"> <li>• <b>Online</b>— PIC is online and running.</li> <li>• <b>Offline</b>—PIC is powered down.</li> </ul>
<b>PIC version</b>	PIC hardware version.
<b>Uptime</b>	How long the PIC has been online.
<b>Port Number</b>	Port number for the PIC.
<b>Cable Type</b>	Type of cable connected to the port: LH, LX, or SX.

Table 29: show chassis pic Output Fields (*continued*)

Field Name	Field Description
<b>PIC Port Information</b>	<p>Port-level information for the PIC.</p> <ul style="list-style-type: none"> <li>• Port—Port number</li> <li>• Cable type—Type of transceiver installed.</li> <li>• Fiber type—Type of fiber.</li> <li>• Xcvr vendor—Transceiver vendor name.</li> <li>• Xcvr vendor part number—Transceiver vendor part number.</li> <li>• Wavelength—Wavelength of the transmitted signal.</li> </ul>

## 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:

  Port  Cable type      Fiber type  Xcvr vendor  Xcvr vendor  Wavelength
      0    10GBASE SR      MM         FINISAR CORP. FTLX8571D3BNL-J1 850 nm

      Xcvr vendor
      firmware version
      0.0

PIC port information:

  Port  Cable type      Fiber type  Xcvr vendor  Xcvr vendor  Wavelength
      1    10GBASE SR      MM         FINISAR CORP. FTLX8571D3BNL-J1 850 nm

      Xcvr vendor
      firmware version
      0.0

```

## show chassis power

**Supported Platforms** [SRX Series, vSRX](#)

**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). This topic is supported on the SRX5400, SRX5600, and SRX5800 devices.

**Options** This command has no options.

**Required Privilege Level** view

**Related Documentation**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [show chassis power sequence on page 363](#)

**List of Sample Output** [show chassis power on page 361](#)

**Output Fields** [Table 30 on page 360](#) lists the output fields for the **show chassis power** command. Output fields are listed in the approximate order in which they appear.

**Table 30: show chassis power Output Fields**

Field Name	Field Description
PEM number	<p>AC or DC PEM number on the chassis. The following output fields are displayed for the PEM:</p> <ul style="list-style-type: none"> <li>• State—State of the PEM: <ul style="list-style-type: none"> <li>• Online—PEM is present in the slot and online.</li> <li>• Empty—PEM is not present in the slot.</li> <li>• Present—PEM is present in the slot, but not online.</li> </ul> </li> <li>• AC Input —State of the AC input power feed with the number of active and expected feeds (1 or 2).</li> <li>• Capacity—Actual power input capacity with maximum capacity displayed (in parentheses) in watts.</li> <li>• 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.</li> </ul>



Table 30: show chassis power Output Fields (*continued*)

Field Name	Field Description
<b>System</b>	<p>Overall power statistics for the system zone:</p> <ul style="list-style-type: none"> <li>• Zone number: <ul style="list-style-type: none"> <li>• Capacity—Maximum power capacity available for the zone, in watts.</li> <li>• Allocated power—Actual capacity allocated for the zone, in watts, with remaining power displayed in parentheses.</li> <li>• Actual usage—Actual power usage for the zone, in watts.</li> </ul> </li> <li>• Total system capacity—Cumulative power capacity of all the zones, in watts.</li> <li>• Total remaining capacity—Difference between the total system capacity and cumulative allocated power of all zones, in watts.</li> </ul>

## Sample Output

### show chassis power

```

user@host> show chassis power

PEM 0:
  State:      Present
  DC input:   Out of range (1 feed expected, 0 feed connected)
  DC input:   48.0 V input (0 mV)
  Capacity:   0 W (maximum 4100 W)

PEM 1:
  State:      Present
  DC input:   Out of range (1 feed expected, 0 feed connected)
  DC input:   48.0 V input (0 mV)
  Capacity:   0 W (maximum 4100 W)

PEM 2:
  State:      Online
  DC input:   OK (2 feed expected, 2 feed connected)
  DC input:   48.0 V input (57500 mV)
  Capacity:   4100 W (maximum 4100 W)
  DC output:  1881 W (zone 0, 33 A at 57 V, 45% of capacity)

PEM 3:
  State:      Online
  DC input:   OK (2 feed expected, 2 feed connected)
  DC input:   48.0 V input (56500 mV)
  Capacity:   4100 W (maximum 4100 W)
  DC output:  2464 W (zone 1, 44 A at 56 V, 60% of capacity)

System:
  Zone 0:
    Capacity:      4100 W (maximum 4100 W)
    Allocated power: 2920 W (1180 W remaining)
    Actual usage:   1881 W
  Zone 1:
    Capacity:      4100 W (maximum 4100 W)
    Allocated power: 3875 W (225 W remaining)
    Actual usage:   2464 W

```

Total system capacity: 8200 W (maximum 8200 W)  
Total remaining power: 1405 W

## show chassis power sequence

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [fru-poweron-sequence on page 243](#)

**List of Sample Output** [show chassis power sequence on page 363](#)

**Output Fields** [Table 31 on page 363](#) lists the output fields for the **show chassis power sequence** command. Output fields are listed in the approximate order in which they appear.

**Table 31: 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)

**Supported Platforms** [SRX Series, vSRX](#)

**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 365](#)

**Output Fields** [Table 32 on page 364](#) lists the output fields for the **show firewall** command. Output fields are listed in the approximate order in which they appear.

**Table 32: show firewall Output Fields**

Field Name	Field Description
<b>Filter</b>	<p>Name of a filter that has been configured with the <b>filter</b> at the <b>[edit firewall]</b> hierarchy level.</p> <p>When an interface-specific filter is displayed, the name of the filter is followed by the full interface name and by either <b>-i</b> for an input filter or <b>-o</b> for an output filter.</p> <p>When dynamic filters are displayed, the name of the filter is followed by the full interface name and by either <b>-in</b> for an input filter or <b>-out</b> 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, <b>__ls1/filter1</b>).</p>

Table 32: show firewall Output Fields (*continued*)

Field Name	Field Description
<b>Counters</b>	<p>Display filter counter information:</p> <ul style="list-style-type: none"> <li>• <b>Name</b>—Name of a filter counter that has been configured with the <b>counter</b> firewall filter action.</li> <li>• <b>Bytes</b>—Number of bytes that match the filter term under which the <b>counter</b> action is specified.</li> <li>• <b>Packets</b>—Number of packets that matched the filter term under which the <b>counter</b> action is specified.</li> </ul>
<b>Policers</b>	<p>Display policer information:</p> <ul style="list-style-type: none"> <li>• <b>Name</b>—Name of policer.</li> <li>• <b>Bytes</b>—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.</li> <li>• <b>Packets</b>—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.</li> </ul>

## 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)

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**
- `aenumber | rethnumber`—(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**

- [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show interfaces extensive \(Aggregated Ethernet\) on page 373](#)

**Output Fields** [Table 33 on page 367](#) lists the output fields for the **show interfaces** (Aggregated Ethernet) command. Output fields are listed in the approximate order in which they appear.

Table 33: Aggregated Ethernet show interfaces Output Fields

Field Name	Field Description	Level of Output
<b>Physical Interface</b>		
<b>Physical interface</b>	Name of the physical interface and state of the interface.	All levels
<b>Enabled</b>	State of the physical interface.	All levels
<b>Interface index</b>	Index number of the physical interface, which reflects its initialization sequence.	All levels
<b>SNMP ifIndex</b>	SNMP index number for the physical interface.	<b>detail extensive</b> none
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
<b>Link-level type</b>	Encapsulation being used on the physical interface.	All levels
<b>MTU</b>	Maximum transmission unit size on the physical interface.	All levels
<b>Speed</b>	Speed at which the interface is running.	All levels
<b>Loopback</b>	Loopback status: <b>Enabled</b> or <b>Disabled</b> . If loopback is enabled, type of loopback: <b>Local</b> or <b>Remote</b> .	All levels
<b>Source filtering</b>	Source filtering status: <b>Enabled</b> or <b>Disabled</b> .	All levels
<b>Flow control</b>	Flow control status: <b>Enabled</b> or <b>Disabled</b> .	All levels
<b>Minimum links needed</b>	Number of child links that must be operational for the aggregate interface to be operational.	All levels
<b>Device flags</b>	Information about the physical device.	All levels
<b>Interface flags</b>	Information about the interface.	All levels
<b>Current address</b>	Configured MAC address.	<b>detail extensive</b>
<b>Hardware address</b>	Hardware MAC address.	<b>detail extensive</b>
<b>Last flapped</b>	Date, time, and how long ago the interface went from down to up or up to down. The format is <b>Last flapped: year-month-day hours:minutes:seconds timezone (hours:minutes:seconds ago)</b> . For example, <b>Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago)</b> .	<b>detail extensive</b>
<b>Input Rate</b>	Input rate in bits per second (bps) and packets per second (pps).	None specified
<b>Output Rate</b>	Output rate in bps and pps.	None specified
<b>Statistics last cleared</b>	Time when the statistics for the interface were last set to zero.	<b>detail extensive</b>

Table 33: Aggregated Ethernet show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Traffic statistics</b>	<p>Number and rate of bytes and packets received and transmitted on the physical interface.</p> <ul style="list-style-type: none"> <li>• <b>Input bytes</b>—Number of bytes received on the interface.</li> <li>• <b>Output bytes</b>—Number of bytes transmitted on the interface.</li> <li>• <b>Input packets</b>—Number of packets received on the interface</li> <li>• <b>Output packets</b>—Number of packets transmitted on the interface.</li> </ul>	<b>detail extensive</b>
<b>IPv6 transit statistics</b>	<p>Number of IPv6 transit bytes and packets received and transmitted on the physical interface if IPv6 statistics tracking is enabled.</p> <ul style="list-style-type: none"> <li>• <b>Input bytes</b>—Number of bytes received on the interface.</li> <li>• <b>Output bytes</b>—Number of bytes transmitted on the interface.</li> <li>• <b>Input packets</b>—Number of packets received on the interface.</li> <li>• <b>Output packets</b>—Number of packets transmitted on the interface.</li> </ul>	<b>detail extensive</b>
<b>Input errors</b>	<p>Input errors on the interface:</p> <ul style="list-style-type: none"> <li>• <b>Errors</b>—Sum of incoming frame aborts and frame check sequence (FCS) errors.</li> <li>• <b>Drops</b>—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.</li> <li>• <b>Framing errors</b>—Number of packets received with an invalid FCS.</li> <li>• <b>Runts</b>—Number of frames received that are smaller than the runt threshold.</li> <li>• <b>Giants</b>—Number of frames received that are larger than the giant threshold.</li> <li>• <b>Policed discards</b>—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.</li> <li>• <b>Resource errors</b>—Sum of transmit drops.</li> </ul>	<b>detail extensive</b>
<b>Output errors</b>	<p>Output errors on the interface. The following paragraphs explain the counters whose meaning might not be obvious:</p> <ul style="list-style-type: none"> <li>• <b>Carrier transitions</b> —Number of times the interface has gone from <b>down</b> to <b>up</b>. 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.</li> <li>• <b>Errors</b>—Sum of the outgoing frame aborts and FCS errors.</li> <li>• <b>Drops</b>—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.</li> <li>• <b>MTU errors</b>—Number of packets whose size exceeded the MTU of the interface.</li> <li>• <b>Resource errors</b>—Sum of transmit drops.</li> </ul>	<b>detail extensive</b>
<b>Egress queues</b>	Total number of egress queues supported on the specified interface	<b>detail extensive</b>



Table 33: Aggregated Ethernet show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Queue counters</b>	CoS queue number and its associated user-configured forwarding class name. <ul style="list-style-type: none"> <li><b>Queued packets</b>—Number of queued packets.</li> <li><b>Transmitted packets</b>—Number of transmitted packets.</li> <li><b>Dropped packets</b>—Number of packets dropped by the ASIC's RED mechanism.</li> </ul>	<b>detail extensive</b>
<b>Logical Interface</b>		
<b>Logical interface</b>	Name of the logical interface.	All levels
<b>Index</b>	Index number of the logical interface (which reflects its initialization sequence).	<b>detail extensive</b> none
<b>SNMP ifIndex</b>	SNMP interface index number of the logical interface.	<b>detail extensive</b> none
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
<b>Flags</b>	Information about the logical interface.	All levels
<b>VLAN-Tag</b>	Tag Protocol Identifier (TPID) and VLAN identifier.	All levels
<b>Demux</b>	IP demultiplexing (demux) value that appears if this interface is used as the demux underlying interface. The output is one of the following: <ul style="list-style-type: none"> <li>Source Family Inet</li> <li>Destination Family Inet</li> </ul>	<b>detail extensive</b> none
<b>Encapsulation</b>	Encapsulation on the logical interface.	All levels
<b>Statistics</b>	Information about the number of packets, packets per second, number of bytes, and bytes per second on this aggregate interface. <ul style="list-style-type: none"> <li><b>Bundle</b>—Information about input and output bundle rates.</li> <li><b>Link</b>—(<b>detail</b> and <b>extensive</b> only) Information about specific links in the aggregate, including link state and input and output rates.</li> </ul>	<b>detail extensive</b> none
<b>LACP info</b>	Link Aggregation Control Protocol (LACP) information for each aggregated interface. <ul style="list-style-type: none"> <li><b>Role</b> can be one of the following: <ul style="list-style-type: none"> <li><b>Actor</b>—Local device participating in LACP negotiation.</li> <li><b>Partner</b>—Remote device participating in LACP negotiation.</li> </ul> </li> <li><b>System priority</b>—Priority assigned to the system (by management or administrative policy), encoded as an unsigned integer.</li> <li><b>System identifier</b>—Actor or partner system ID, encoded as a MAC address.</li> <li><b>Port priority</b>—Priority assigned to the port by the actor or partner (by management or administrative policy), encoded as an unsigned integer.</li> <li><b>Port number</b>—Port number assigned to the port by the actor or partner, encoded as an unsigned integer.</li> <li><b>Port key</b>—Operational key value assigned to the port by the actor or partner, encoded as an unsigned integer.</li> </ul>	<b>detail extensive</b> none

Table 33: Aggregated Ethernet show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>LACP Statistics</b>	<p>LACP statistics for each aggregated interface.</p> <ul style="list-style-type: none"> <li>• <b>LACP Rx</b>—LACP received counter that increments for each normal hello.</li> <li>• <b>LACP Tx</b>—Number of LACP transmit packet errors logged.</li> <li>• <b>Unknown Rx</b>—Number of unrecognized packet errors logged.</li> <li>• <b>Illegal Rx</b>—Number of invalid packets received.</li> </ul>	<b>detail extensive none</b>
<b>Marker Statistic</b>	<p>(<b>detail</b> and <b>extensive</b> only) Information about 802.3ad marker protocol statistics on the specified links.</p> <ul style="list-style-type: none"> <li>• <b>Marker Rx</b>—Number of valid marker PDUs received on this aggregation port.</li> <li>• <b>Resp Tx</b>—Number of marker response PDUs transmitted on this aggregation port.</li> <li>• <b>Unknown Rx</b>—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.</li> <li>• <b>Illegal Rx</b>—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).</li> </ul>	<b>detail extensive none</b>
<b>Flow Statistics</b>	Flow statistics for each aggregated interface.	<b>detail extensive none</b>
<b>Flow Input statistics</b>	Statistics for packets received by the flow module.	<b>detail extensive none</b>
<b>Flow Output statistics</b>	Statistics for packets sent by the flow module.	<b>detail extensive none</b>

Table 33: Aggregated Ethernet show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Flow error statistics (Packets dropped due to)</b>	<p>Packet drop statistics for the flow module.</p> <ul style="list-style-type: none"> <li>• <b>Address spoofing</b>—Packet dropped when the screen module detected address spoofing.</li> <li>• <b>Authentication failed</b>—Packet dropped because the IPsec Encapsulating Security Payload (ESP) or Authentication Header (AH) authentication failed.</li> <li>• <b>Incoming NAT errors</b>—Packet dropped because the source NAT rule search failed, an invalid source NAT binding was found, or the NAT allocation failed.</li> <li>• <b>Invalid zone received packet</b>—This counter is not currently in use.</li> <li>• <b>Multiple user authentications</b>—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.)</li> <li>• <b>Multiple incoming NAT</b>—Packet dropped if source NAT is specified more than once. (Sometimes packets are looped through the system more than once.)</li> <li>• <b>No parent for a gate</b>—This counter is not currently in use.</li> <li>• <b>No one interested in self packets</b>—This counter is incremented for one of the following reasons: <ul style="list-style-type: none"> <li>• 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.</li> <li>• No service is interested in the to-self packet</li> <li>• 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.</li> </ul> </li> </ul>	<b>detail extensive none</b>

Table 33: Aggregated Ethernet show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Flow error statistics (Packets dropped due to)</b>	Packet drop statistics for the flow module (continued). <ul style="list-style-type: none"> <li>• <b>No minor session</b>—Packet dropped because no minor sessions are available and a minor session was requested. Minor sessions are allocated for storing additional TCP state information.</li> <li>• <b>No more sessions</b>—Packet dropped because there were no more free sessions available.</li> <li>• <b>No NAT gate</b>—This counter is not currently in use.</li> <li>• <b>No route present</b>—Packet dropped because a valid route was not available to forward the packet.</li> <li>• <b>No SA for incoming SPI</b>—Packet dropped because the incoming IPsec packet's security parameter index (SPI) does not match any known SPI.</li> <li>• <b>No tunnel found</b>—Packet dropped because a valid tunnel could not be found.</li> <li>• <b>No session for a gate</b>—Packet dropped by an ALG.</li> <li>• <b>No zone or NULL zone binding</b>—Packet dropped because its incoming interface was not bound to any zone.</li> <li>• <b>Policy denied</b>—The error counter is incremented for one of the following reasons:               <ul style="list-style-type: none"> <li>• Source or destination NAT (or both) has occurred and policy says to drop the packet.</li> <li>• Policy specifies user authentication, which failed.</li> <li>• Policy was configured to deny this packet.</li> </ul> </li> <li>• <b>Security association not active</b>—Packet dropped because an IPsec packet was received for an inactive SA.</li> <li>• <b>TCP sequence number out of window</b>—TCP packet with a sequence number failed the TCP sequence number check that was received.</li> <li>• <b>Syn-attack protection</b>—Packet dropped because of SYN attack protection or SYN cookie protection.</li> <li>• <b>User authentication errors</b>—Packet dropped because policy requires authentication; however:               <ul style="list-style-type: none"> <li>• Only Telnet, FTP, and HTTP traffic can be authenticated.</li> <li>• The corresponding authentication entry could not be found, if web-auth is specified.</li> <li>• The maximum number of authenticated sessions per user was exceeded.</li> </ul> </li> </ul>	<b>detail extensive none</b>
<b>protocol-family</b>	Protocol family configured on the logical interface.	<b>brief</b>
<b>Protocol</b>	Protocol family configured on the logical interface.	<b>detail extensive none</b>
<b>MTU</b>	Maximum transmission unit size on the logical interface.	<b>detail extensive none</b>
<b>Maximum labels</b>	Maximum number of MPLS labels configured for the MPLS protocol family on the logical interface.	<b>detail extensive none</b>
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
<b>Route Table</b>	Routing table in which the logical interface address is located. For example, 0 refers to the routing table inet.0.	<b>detail extensive</b>

Table 33: Aggregated Ethernet show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Flags</b>	Information about protocol family flags.	<b>detail extensive none</b>
<b>Mac-Validate Failures</b>	Number of MAC address validation failures for packets and bytes. This field is displayed when MAC address validation is enabled for the logical interface.	<b>detail extensive none</b>
<b>Addresses, Flags</b>	Information about address flags.	<b>detail extensive none</b>
<b>Destination</b>	IP address of the remote side of the connection.	<b>detail extensive none</b>
<b>Local</b>	IP address of the logical interface.	<b>detail extensive none</b>
<b>Broadcast</b>	Broadcast address of the logical interface.	<b>detail extensive none</b>
<b>Policer</b>	Policer to be evaluated when packets are received or transmitted on the interface.	<b>detail extensive none</b>
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>

## 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 (SRX Series)

**Supported Platforms** SRX Series, vSRX

**Syntax** `show interfaces (`  
     `<interface-name>`  
     `<brief | detail | extensive | terse>`  
     `<controller interface-name>|`  
     `<descriptions interface-name>|`  
     `<destination-class (all | destination-class-name logical-interface-name)>|`  
     `<diagnostics optics interface-name>|`  
     `<far-end-interval interface-fpc/pic/port>|`  
     `<filters interface-name>|`  
     `<flow-statistics interface-name>|`  
     `<interval interface-name>|`  
     `<load-balancing (detail | interface-name)>|`  
     `<mac-database mac-address mac-address>|`  
     `<mc-ae id identifier unit number revertive-info>|`  
     `<media interface-name>|`  
     `<policers interface-name>|`  
     `<queue both-ingress-egress egress forwarding-class forwarding-class ingress l2-statistics>|`  
     `<redundancy (detail | interface-name)>|`  
     `<routing brief detail summary interface-name>|`  
     `<routing-instance (all | instance-name)>|`  
     `<snmp-index snmp-index>|`  
     `<source-class (all | destination-class-name logical-interface-name)>|`  
     `<statistics interface-name>|`  
     `<switch-port switch-port number>|`  
     `<transport pm (all | optics | otn) (all | current | currentday | interval | previousday) (all |`  
         `interface-name)>|`  
     `<zone interface-name>`  
     `)`

**Release Information** Command modified in Junos OS Release 9.5.

**Description** Display status information and statistics about interfaces on SRX Series appliance running Junos OS.

On SRX Series appliance, on configuring identical IPs on a single interface, you will not see a warning message; instead, you will see a syslog message.

- Options**
- **interface-name**—(Optional) Display standard information about the specified interface. Following is a list of typical interface names. Replace pim with the PIM slot and port with the port number.
    - **at-*pim*/0/*port***—ATM-over-ADSL or ATM-over-SHDSL interface.
    - **ce1-*pim*/0/ *port***—Channelized E1 interface.
    - **cl-0/0/8**—3G wireless modem interface for SRX320 devices.
    - **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).
- 
- **brief | detail | extensive | terse**—(Optional) Display the specified level of output.
  - **controller**—(Optional) Show controller information.
  - **descriptions**—(Optional) Display interface description strings.
  - **destination-class**—(Optional) Show statistics for destination class.
  - **diagnostics**—(Optional) Show interface diagnostics information.
  - **far-end-interval**—(Optional) Show far end interval statistics.
  - **filters**—(Optional) Show interface filters information.
  - **flow-statistics**—(Optional) Show security flow counters and errors.
  - **interval**—(Optional) Show interval statistics.
  - **load-balancing**—(Optional) Show load-balancing status.
  - **mac-database**—(Optional) Show media access control database information.
  - **mc-ae**—(Optional) Show MC-AE configured interface information.
  - **media**—(Optional) Display media information.
  - **policers**—(Optional) Show interface policers information.
  - **queue**—(Optional) Show queue statistics for this interface.
  - **redundancy**—(Optional) Show redundancy status.
  - **routing**—(Optional) Show routing status.
  - **routing-instance**—(Optional) Name of routing instance.
  - **snmp-index**—(Optional) SNMP index of interface.
  - **source-class**—(Optional) Show statistics for source class.
  - **statistics**—(Optional) Display statistics and detailed output.
  - **switch-port**—(Optional) Front end port number (0..15).

- **transport**—(Optional) Show interface transport information.
- **zone**—(Optional) Interface's zone.

**Required Privilege Level** view

**Related Documentation** • *Understanding Layer 2 Interfaces*

**List of Sample Output**

- [show interfaces Gigabit Ethernet on page 387](#)
- [show interfaces brief \(Gigabit Ethernet\) on page 387](#)
- [show interfaces detail \(Gigabit Ethernet\) on page 388](#)
- [show interfaces statistics st0.0 detail on page 389](#)
- [show interfaces extensive \(Gigabit Ethernet\) on page 390](#)
- [show interfaces terse on page 393](#)
- [show interfaces controller \(Channelized E1 IQ with Logical E1\) on page 393](#)
- [show interfaces controller \(Channelized E1 IQ with Logical DSO\) on page 394](#)
- [show interfaces descriptions on page 394](#)
- [show interfaces destination-class all on page 394](#)
- [show interfaces diagnostics optics on page 394](#)
- [show interfaces far-end-interval coc12-5/2/0 on page 395](#)
- [show interfaces far-end-interval coc1-5/2/1:1 on page 396](#)
- [show interfaces filters on page 396](#)
- [show interfaces flow-statistics \(Gigabit Ethernet\) on page 396](#)
- [show interfaces interval \(Channelized OC12\) on page 397](#)
- [show interfaces interval \(E3\) on page 398](#)
- [show interfaces interval \(SONET/SDH\) on page 398](#)
- [show interfaces load-balancing on page 399](#)
- [show interfaces load-balancing detail on page 399](#)
- [show interfaces mac-database \(All MAC Addresses on a Port\) on page 399](#)
- [show interfaces mac-database \(All MAC Addresses on a Service\) on page 400](#)
- [show interfaces mac-database mac-address on page 400](#)
- [show interfaces mc-ae on page 401](#)
- [show interfaces media \(SONET/SDH\) on page 401](#)
- [show interfaces policers on page 401](#)
- [show interfaces policers interface-name on page 402](#)
- [show interfaces queue on page 402](#)
- [show interfaces redundancy on page 403](#)
- [show interfaces redundancy \(Aggregated Ethernet\) on page 403](#)
- [show interfaces redundancy detail on page 403](#)
- [show interfaces routing brief on page 403](#)
- [show interfaces routing detail on page 404](#)
- [show interfaces routing-instance all on page 404](#)
- [show interfaces snmp-index on page 405](#)
- [show interfaces source-class all on page 405](#)
- [show interfaces statistics \(Fast Ethernet\) on page 405](#)
- [show interfaces switch-port on page 406](#)
- [show interfaces transport pm on page 407](#)

[show security zones on page 408](#)

**Output Fields** [Table 34 on page 380](#) lists the output fields for the **show interfaces** command. Output fields are listed in the approximate order in which they appear.

**Table 34: show interfaces Output Fields**

Field Name	Field Description	Level of Output
<b>Physical Interface</b>		
<b>Physical interface</b>	Name of the physical interface.	All levels
<b>Enabled</b>	State of the interface.	All levels
<b>Interface index</b>	Index number of the physical interface, which reflects its initialization sequence.	<b>detail extensive none</b>
<b>SNMP ifIndex</b>	SNMP index number for the physical interface.	<b>detail extensive none</b>
<b>Link-level type</b>	Encapsulation being used on the physical interface.	All levels
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
<b>MTU</b>	Maximum transmission unit size on the physical interface.	All levels
<b>Link mode</b>	Link mode: Full-duplex or Half-duplex.	
<b>Speed</b>	Speed at which the interface is running.	All levels
<b>BPDU error</b>	Bridge protocol data unit (BPDU) error: Detected or None	
<b>Loopback</b>	Loopback status: <b>Enabled</b> or <b>Disabled</b> . If loopback is enabled, type of loopback: <b>Local</b> or <b>Remote</b> .	All levels
<b>Source filtering</b>	Source filtering status: <b>Enabled</b> or <b>Disabled</b> .	All levels
<b>Flow control</b>	Flow control status: <b>Enabled</b> or <b>Disabled</b> .	All levels
<b>Auto-negotiation</b>	(Gigabit Ethernet interfaces) Autonegotiation status: <b>Enabled</b> or <b>Disabled</b> .	All levels
<b>Remote-fault</b>	(Gigabit Ethernet interfaces) Remote fault status: <ul style="list-style-type: none"> <li>• <b>Online</b>—Autonegotiation is manually configured as online.</li> <li>• <b>Offline</b>—Autonegotiation is manually configured as offline.</li> </ul>	All levels
<b>Device flags</b>	Information about the physical device.	All levels
<b>Interface flags</b>	Information about the interface.	All levels
<b>Link flags</b>	Information about the physical link.	All levels
<b>CoS queues</b>	Number of CoS queues configured.	<b>detail extensive none</b>

Table 34: show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
Current address	Configured MAC address.	detail extensive none
Last flapped	Date, time, and how long ago the interface went from down to up. The format is <b>Last flapped: year-month-day hour:minute:second:timezone (hour:minute:second ago)</b> . For example, <b>Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago)</b> .	detail extensive none
Input Rate	Input rate in bits per second (bps) and packets per second (pps).	None
Output Rate	Output rate in bps and pps.	None
Active alarms and Active defects	<p>Ethernet-specific defects that can prevent the interface from passing packets. When a defect persists for a certain amount of time, it is promoted to an alarm. These fields can contain the value <b>None</b> or <b>Link</b>.</p> <ul style="list-style-type: none"> <li>• <b>None</b>—There are no active defects or alarms.</li> <li>• <b>Link</b>—Interface has lost its link state, which usually means that the cable is unplugged, the far-end system has been turned off, or the PIC is malfunctioning.</li> </ul>	detail extensive none
Statistics last cleared	Time when the statistics for the interface were last set to zero.	detail extensive
Traffic statistics	<p>Number and rate of bytes and packets received and transmitted on the physical interface.</p> <ul style="list-style-type: none"> <li>• <b>Input bytes</b>—Number of bytes received on the interface.</li> <li>• <b>Output bytes</b>—Number of bytes transmitted on the interface.</li> <li>• <b>Input packets</b>—Number of packets received on the interface.</li> <li>• <b>Output packets</b>—Number of packets transmitted on the interface.</li> </ul>	detail extensive

Table 34: show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Input errors</b>	<p>Input errors on the interface.</p> <ul style="list-style-type: none"> <li>• <b>Errors</b>—Sum of the incoming frame aborts and FCS errors.</li> <li>• <b>Drops</b>—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.</li> <li>• <b>Framing errors</b>—Number of packets received with an invalid frame checksum (FCS).</li> <li>• <b>Runts</b>—Number of frames received that are smaller than the runt threshold.</li> <li>• <b>Policed discards</b>—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 Junos OS does not handle.</li> <li>• <b>L3 incompletes</b>—Number of incoming packets discarded because they failed Layer 3 (usually IPv4) sanity checks of the header. For example, a frame with less than 20 bytes of available IP header is discarded. L3 incomplete errors can be ignored by configuring the <b>ignore-l3-incompletes</b>.</li> <li>• <b>L2 channel errors</b>—Number of times the software did not find a valid logical interface for an incoming frame.</li> <li>• <b>L2 mismatch timeouts</b>—Number of malformed or short packets that caused the incoming packet handler to discard the frame as unreadable.</li> <li>• <b>FIFO errors</b>—Number of FIFO errors in the receive direction that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning.</li> <li>• <b>Resource errors</b>—Sum of transmit drops.</li> </ul>	<b>extensive</b>
<b>Output errors</b>	<p>Output errors on the interface.</p> <ul style="list-style-type: none"> <li>• <b>Carrier transitions</b>—Number of times the interface has gone from <b>down</b> to <b>up</b>. 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 or PIM is malfunctioning.</li> <li>• <b>Errors</b>—Sum of the outgoing frame aborts and FCS errors.</li> <li>• <b>Drops</b>—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.</li> <li>• <b>Collisions</b>—Number of Ethernet collisions. The Gigabit Ethernet PIC supports only full-duplex operation; therefore, for Gigabit Ethernet PICs, this number must always remain 0. If it is nonzero, there is a software bug.</li> <li>• <b>Aged packets</b>—Number of packets that remained in shared packet SDRAM so long that the system automatically purged them. The value in this field must never increment. If it does, it is most likely a software bug or possibly malfunctioning hardware.</li> <li>• <b>FIFO errors</b>—Number of FIFO errors in the send direction as reported by the ASIC on the PIC. If this value is ever nonzero, the PIC is probably malfunctioning.</li> <li>• <b>HS link CRC errors</b>—Number of errors on the high-speed links between the ASICs responsible for handling the interfaces.</li> <li>• <b>MTU errors</b>—Number of packets whose size exceeded the MTU of the interface.</li> <li>• <b>Resource errors</b>—Sum of transmit drops.</li> </ul>	<b>extensive</b>

Table 34: show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Ingress queues</b>	Total number of ingress queues supported on the specified interface.	<b>extensive</b>
<b>Queue counters and queue number</b>	CoS queue number and its associated user-configured forwarding class name. <ul style="list-style-type: none"> <li>• <b>Queued packets</b>—Number of queued packets.</li> <li>• <b>Transmitted packets</b>—Number of transmitted packets.</li> <li>• <b>Dropped packets</b>—Number of packets dropped by the ASIC's RED mechanism.</li> </ul>	<b>detail extensive</b>
<b>MAC statistics</b>	<p>Receive and Transmit statistics reported by the PIC's MAC subsystem, including the following:</p> <ul style="list-style-type: none"> <li>• <b>Total octets and total packets</b>—Total number of octets and packets.</li> <li>• <b>Unicast packets, Broadcast packets, and Multicast packets</b>—Number of unicast, broadcast, and multicast packets.</li> <li>• <b>CRC/Align errors</b>—Total number of packets received that had a length (excluding framing bits, but including FCS octets) of between 64 and 1518 octets, inclusive, and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error).</li> <li>• <b>FIFO error</b>—Number of FIFO errors that are reported by the ASIC on the PIC. If this value is ever nonzero, the PIC or a cable is probably malfunctioning.</li> <li>• <b>MAC control frames</b>—Number of MAC control frames.</li> <li>• <b>MAC pause frames</b>—Number of MAC control frames with <b>pause</b> operational code.</li> <li>• <b>Oversized frames</b>—There are two possible conditions regarding the number of oversized frames: <ul style="list-style-type: none"> <li>• Packet length exceeds 1518 octets, or</li> <li>• Packet length exceeds MRU</li> </ul> </li> <li>• <b>Jabber frames</b>—Number of frames that were longer than 1518 octets (excluding framing bits, but including FCS octets), and had either an FCS error or an alignment error. This definition of jabber is different from the definition in IEEE-802.3 section 8.2.1.5 (10BASE5) and section 10.3.1.4 (10BASE2). These documents define jabber as the condition in which any packet exceeds 20 ms. The allowed range to detect jabber is from 20 ms to 150 ms.</li> <li>• <b>Fragment frames</b>—Total number of packets that were less than 64 octets in length (excluding framing bits, but including FCS octets) and had either an FCS error or an alignment error. Fragment frames normally increment because both runts (which are normal occurrences caused by collisions) and noise hits are counted.</li> <li>• <b>VLAN tagged frames</b>—Number of frames that are VLAN tagged. The system uses the TPID of 0x8100 in the frame to determine whether a frame is tagged or not.</li> <li>• <b>Code violations</b>—Number of times an event caused the PHY to indicate "Data reception error" or "invalid data symbol error."</li> </ul>	<b>extensive</b>

Table 34: show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
Filter statistics	<p>Receive and Transmit statistics reported by the PIC's MAC address filter subsystem. The filtering is done by the content-addressable memory (CAM) on the PIC. The filter examines a packet's source and destination MAC addresses to determine whether the packet should enter the system or be rejected.</p> <ul style="list-style-type: none"> <li>• <b>Input packet count</b>—Number of packets received from the MAC hardware that the filter processed.</li> <li>• <b>Input packet rejects</b>—Number of packets that the filter rejected because of either the source MAC address or the destination MAC address.</li> <li>• <b>Input DA rejects</b>—Number of packets that the filter rejected because the destination MAC address of the packet is not on the accept list. It is normal for this value to increment. When it increments very quickly and no traffic is entering the device from the far-end system, either there is a bad ARP entry on the far-end system, or multicast routing is not on and the far-end system is sending many multicast packets to the local device (which the router is rejecting).</li> <li>• <b>Input SA rejects</b>—Number of packets that the filter rejected because the source MAC address of the packet is not on the accept list. The value in this field should increment only if source MAC address filtering has been enabled. If filtering is enabled, if the value increments quickly, and if the system is not receiving traffic that it should from the far-end system, it means that the user-configured source MAC addresses for this interface are incorrect.</li> <li>• <b>Output packet count</b>—Number of packets that the filter has given to the MAC hardware.</li> <li>• <b>Output packet pad count</b>—Number of packets the filter padded to the minimum Ethernet size (60 bytes) before giving the packet to the MAC hardware. Usually, padding is done only on small ARP packets, but some very small IP packets can also require padding. If this value increments rapidly, either the system is trying to find an ARP entry for a far-end system that does not exist or it is misconfigured.</li> <li>• <b>Output packet error count</b>—Number of packets with an indicated error that the filter was given to transmit. These packets are usually aged packets or are the result of a bandwidth problem on the FPC hardware. On a normal system, the value of this field should not increment.</li> <li>• <b>CAM destination filters, CAM source filters</b>—Number of entries in the CAM dedicated to destination and source MAC address filters. There can only be up to 64 source entries. If source filtering is disabled, which is the default, the values for these fields must be 0.</li> </ul>	extensive
Autonegotiation information	<p>Information about link autonegotiation.</p> <ul style="list-style-type: none"> <li>• <b>Negotiation status:</b> <ul style="list-style-type: none"> <li>• <b>Incomplete</b>—Ethernet interface has the speed or link mode configured.</li> <li>• <b>No autonegotiation</b>—Remote Ethernet interface has the speed or link mode configured, or does not perform autonegotiation.</li> <li>• <b>Complete</b>—Ethernet interface is connected to a device that performs autonegotiation and the autonegotiation process is successful.</li> </ul> </li> </ul>	extensive
Packet Forwarding Engine configuration	<p>Information about the configuration of the Packet Forwarding Engine:</p> <ul style="list-style-type: none"> <li>• <b>Destination slot</b>—FPC slot number.</li> </ul>	extensive



Table 34: show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>CoS information</b>	Information about the CoS queue for the physical interface. <ul style="list-style-type: none"> <li>• <b>CoS transmit queue</b>—Queue number and its associated user-configured forwarding class name.</li> <li>• <b>Bandwidth %</b>—Percentage of bandwidth allocated to the queue.</li> <li>• <b>Bandwidth bps</b>—Bandwidth allocated to the queue (in bps).</li> <li>• <b>Buffer %</b>—Percentage of buffer space allocated to the queue.</li> <li>• <b>Buffer usec</b>—Amount of buffer space allocated to the queue, in microseconds. This value is nonzero only if the buffer size is configured in terms of time.</li> <li>• <b>Priority</b>—Queue priority: <b>low</b> or <b>high</b>.</li> <li>• <b>Limit</b>—Displayed if rate limiting is configured for the queue. Possible values are <b>none</b> and <b>exact</b>. If <b>exact</b> is configured, the queue transmits only up to the configured bandwidth, even if excess bandwidth is available. If <b>none</b> is configured, the queue transmits beyond the configured bandwidth if bandwidth is available.</li> </ul>	<b>extensive</b>
<b>Interface transmit statistics</b>	Status of the <b>interface-transmit-statistics</b> configuration: Enabled or Disabled.	<b>detail extensive</b>
<b>Queue counters (Egress)</b>	CoS queue number and its associated user-configured forwarding class name. <ul style="list-style-type: none"> <li>• <b>Queued packets</b>—Number of queued packets.</li> <li>• <b>Transmitted packets</b>—Number of transmitted packets.</li> <li>• <b>Dropped packets</b>—Number of packets dropped by the ASIC's RED mechanism.</li> </ul>	<b>detail extensive</b>
<b>Logical Interface</b>		
<b>Logical interface</b>	Name of the logical interface.	All levels
<b>Index</b>	Index number of the logical interface, which reflects its initialization sequence.	<b>detail extensive none</b>
<b>SNMP ifIndex</b>	SNMP interface index number for the logical interface.	<b>detail extensive none</b>
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
<b>Flags</b>	Information about the logical interface.	All levels
<b>Encapsulation</b>	Encapsulation on the logical interface.	All levels
<b>Traffic statistics</b>	Number and rate of bytes and packets received and transmitted on the specified interface set. <ul style="list-style-type: none"> <li>• <b>Input bytes, Output bytes</b>—Number of bytes received and transmitted on the interface set. The value in this field also includes the Layer 2 overhead bytes for ingress or egress traffic on Ethernet interfaces if you enable accounting of Layer 2 overhead at the PIC level or the logical interface level.</li> <li>• <b>Input packets, Output packets</b>—Number of packets received and transmitted on the interface set.</li> </ul>	<b>detail extensive</b>

Table 34: show interfaces Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Local statistics</b>	Number and rate of bytes and packets destined to the device.	<b>extensive</b>
<b>Transit statistics</b>	Number and rate of bytes and packets transiting the switch.  <b>NOTE:</b> For Gigabit Ethernet intelligent queuing 2 (IQ2) interfaces, the logical interface egress statistics might not accurately reflect the traffic on the wire when output shaping is applied. Traffic management output shaping might drop packets after they are tallied by the <b>Output bytes</b> and <b>Output packets</b> interface counters. However, correct values display for both of these egress statistics when per-unit scheduling is enabled for the Gigabit Ethernet IQ2 physical interface, or when a single logical interface is actively using a shared scheduler.	<b>extensive</b>
<b>Security</b>	Security zones that interface belongs to.	<b>extensive</b>
<b>Flow Input statistics</b>	Statistics on packets received by flow module.	<b>extensive</b>
<b>Flow Output statistics</b>	Statistics on packets sent by flow module.	<b>extensive</b>
<b>Flow error statistics (Packets dropped due to)</b>	Statistics on errors in the flow module.	<b>extensive</b>
<b>Protocol</b>	Protocol family.	<b>detail extensive none</b>
<b>MTU</b>	Maximum transmission unit size on the logical interface.	<b>detail extensive none</b>
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>
<b>Route Table</b>	Route table in which the logical interface address is located. For example, 0 refers to the routing table inet.0.	<b>detail extensive none</b>
<b>Flags</b>	Information about protocol family flags. .	<b>detail extensive</b>
<b>Addresses, Flags</b>	Information about the address flags..	<b>detail extensive none</b>
<b>Destination</b>	IP address of the remote side of the connection.	<b>detail extensive none</b>
<b>Local</b>	IP address of the logical interface.	<b>detail extensive none</b>
<b>Broadcast</b>	Broadcast address of the logical interface.	<b>detail extensive none</b>
<b>Generation</b>	Unique number for use by Juniper Networks technical support only.	<b>detail extensive</b>

## Sample Output

### show interfaces Gigabit Ethernet

```

user@host> show interfaces ge-0/0/1
Physical interface: ge-0/0/1, Enabled, Physical link is Down
  Interface index: 135, SNMP ifIndex: 510
  Link-level type: Ethernet, MTU: 1514, Link-mode: Full-duplex, Speed: 1000mbps,

  BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:1f:12:e4:b1:01, Hardware address: 00:1f:12:e4:b1:01
  Last flapped   : 2015-05-12 08:36:59 UTC (1w1d 22:42 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : LINK
  Active defects : LINK
  Interface transmit statistics: Disabled

Logical interface ge-0/0/1.0 (Index 71) (SNMP ifIndex 514)
  Flags: Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
  Input packets : 0
  Output packets: 0
  Security: Zone: public
  Protocol inet, MTU: 1500
    Flags: Sendbroadcast-pkt-to-re
    Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
      Destination: 1.1.1/24, Local: 1.1.1.1, Broadcast: 1.1.1.255

```

## Sample Output

### show interfaces brief (Gigabit Ethernet)

```

user@host> show interfaces ge-3/0/2 brief
Physical interface: ge-3/0/2, Enabled, Physical link is Up
  Link-level type: 52, MTU: 1522, Speed: 1000mbps, Loopback: Disabled,
  Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None

Logical interface ge-3/0/2.0
  Flags: SNMP-Traps 0x4000
  VLAN-Tag [ 0x8100.512 0x8100.513 ] In(pop-swap 0x8100.530) Out(swap-push
0x8100.512 0x8100.513)
  Encapsulation: VLAN-CCC
  ccc

Logical interface ge-3/0/2.32767
  Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x0000.0 ] Encapsulation: ENET2

```

## Sample Output

### show interfaces detail (Gigabit Ethernet)

```

user@host> show interfaces ge-0/0/1 detail
Physical interface: ge-0/0/1, Enabled, Physical link is Down
  Interface index: 135, SNMP ifIndex: 510, Generation: 138
  Link-level type: Ethernet, MTU: 1514, Link-mode: Full-duplex, Speed: 1000mbps,
  BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled, Source filtering:
  Disabled,
  Flow control: Enabled, Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues    : 8 supported, 8 maximum usable queues
  Hold-times    : Up 0 ms, Down 0 ms
  Current address: 00:1f:12:e4:b1:01, Hardware address: 00:1f:12:e4:b1:01
  Last flapped  : 2015-05-12 08:36:59 UTC (1w2d 00:00 ago)
  Statistics last cleared: Never
  Traffic statistics:
    Input bytes  :                0                0 bps
    Output bytes :                0                0 bps
    Input packets:                0                0 pps
    Output packets:              0                0 pps
  Egress queues: 8 supported, 4 in use
  Queue counters:
    Queued packets  Transmitted packets  Dropped packets

    0 best-effort           0                0                0
    1 expedited-fo         0                0                0
    2 assured-forw         0                0                0
    3 network-cont         0                0                0

  Queue number:      Mapped forwarding classes
    0                best-effort
    1                expedited-forwarding
    2                assured-forwarding
    3                network-control
  Active alarms  : LINK
  Active defects : LINK
  Interface transmit statistics: Disabled

  Logical interface ge-0/0/1.0 (Index 71) (SNMP ifIndex 514) (Generation 136)
  Flags: Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
  Traffic statistics:
    Input bytes  :                0
    Output bytes :                0
    Input packets:                0
    Output packets:              0
  Local statistics:
    Input bytes  :                0
    Output bytes :                0
    Input packets:                0
    Output packets:              0
  Transit statistics:
    Input bytes  :                0                0 bps
    Output bytes :                0                0 bps
    Input packets:                0                0 pps

```

```

Output packets:                                0                0 pps
Security: Zone: public
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 inet, MTU: 1500, Generation: 150, Route table: 0
Flags: Sendbcst-pkt-to-re
Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
Destination: 1.1.1/24, Local: 1.1.1.1, Broadcast: 1.1.1.255, Generation:
150

```

### show interfaces statistics st0.0 detail

```

user@host> show interfaces statistics st0.0 detail
Logical interface st0.0 (Index 71) (SNMP ifIndex 609) (Generation 136)
Flags: Up Point-To-Point SNMP-Traps Encapsulation: Secure-Tunnel
Traffic statistics:
  Input bytes :                               528152756774
  Output bytes :                              575950643520
  Input packets:                             11481581669
  Output packets:                             12520666095
Local statistics:
  Input bytes :                               0
  Output bytes :                              0
  Input packets:                              0
  Output packets:                             0
Transit statistics:
  Input bytes :                               0                121859888 bps
  Output bytes :                              0                128104112 bps
  Input packets:                              0                331141 pps
  Output packets:                             0                348108 pps

```

```

Security: Zone: untrust
Allowed host-inbound traffic : any-service bfd bgp dvmrp igmp ldp msdp nhrp
ospf ospf3 pgm pim rip ripng router-discovery rsvp
sap vrrp
Flow Statistics :
Flow Input statistics :
  Self packets : 0
  ICMP packets : 0
  VPN packets : 0
  Multicast packets : 0
  Bytes permitted by policy : 525984295844
  Connections established : 7
Flow Output statistics:
  Multicast packets : 0
  Bytes permitted by policy : 576003290222
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: 2000280
  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: 9192
Max nh cache: 0, New hold nh limit: 0, Curr nh cnt: 0, Curr new hold cnt: 0,
NH drop cnt: 0
Generation: 155, Route table: 0
Flags: Sendbroadcast-pkt-to-re

```

## Sample Output

### show interfaces extensive (Gigabit Ethernet)

```

user@host> show interfaces ge-0/0/1.0 extensive
Physical interface: ge-0/0/1, Enabled, Physical link is Down
Interface index: 135, SNMP ifIndex: 510, Generation: 138
Link-level type: Ethernet, MTU: 1514, Link-mode: Full-duplex, Speed: 1000mbps,

BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled,
Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
Remote fault: Online
Device flags : Present Running Down
Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
Link flags : None
CoS queues : 8 supported, 8 maximum usable queues
Hold-times : Up 0 ms, Down 0 ms

```

```

Current address: 00:1f:12:e4:b1:01, Hardware address: 00:1f:12:e4:b1:01
Last flapped   : 2015-05-12 08:36:59 UTC (1w1d 22:57 ago)
Statistics last cleared: Never
Traffic statistics:
Input bytes   :                0                0 bps
Output bytes  :                0                0 bps
Input packets :                0                0 pps
Output packets:                0                0 pps
Input errors:
  Errors: 0, Drops: 0, Framing errors: 0, Runt: 0, Policed discards: 0,
  L3 incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0,
  FIFO errors: 0, Resource errors: 0
Output errors:
  Carrier transitions: 0, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,

  FIFO errors: 0, HS link CRC errors: 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           0                0                0
  1 expedited-fo         0                0                0
  2 assured-forw         0                0                0
  3 network-cont         0                0                0

Queue number:      Mapped forwarding classes
0                  best-effort
1                  expedited-forwarding
2                  assured-forwarding
3                  network-control

Active alarms : LINK
Active defects : LINK
MAC statistics:
  Receive          Transmit
Total octets      0                0
Total packets    0                0
Unicast packets  0                0
Broadcast packets 0                0
Multicast packets 0                0
CRC/Align errors 0                0
FIFO errors       0                0
MAC control frames 0                0
MAC pause frames  0                0
Oversized frames  0
Jabber frames     0
Fragment frames   0
VLAN tagged frames 0
Code violations    0
Filter statistics:
Input packet count      0
Input packet rejects    0
Input DA rejects        0
Input SA rejects        0
Output packet count     0
Output packet pad count 0
Output packet error count 0
CAM destination filters: 2, CAM source filters: 0
Autonegotiation information:
  Negotiation status: Incomplete
Packet Forwarding Engine configuration:

```

```

    Destination slot: 0
    CoS information:
    Direction : Output
    CoS transmit queue
Limit      Bandwidth      Buffer Priority
           %      bps      %      usec
    0 best-effort      95      950000000      95      0      low
none
    3 network-control      5      50000000      5      0      low
none
    Interface transmit statistics: Disabled

```

Logical interface ge-0/0/1.0 (Index 71) (SNMP ifIndex 514) (Generation 136)

Flags: Device-Down SNMP-Traps 0x0 Encapsulation: ENET2

Traffic statistics:

```

    Input bytes :      0
    Output bytes :      0
    Input packets:      0
    Output packets:      0

```

Local statistics:

```

    Input bytes :      0
    Output bytes :      0
    Input packets:      0
    Output packets:      0

```

Transit statistics:

```

    Input bytes :      0      0 bps
    Output bytes :      0      0 bps
    Input packets:      0      0 pps
    Output packets:      0      0 pps

```

Security: Zone: public

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 inet, MTU: 1500, Generation: 150, Route table: 0
Flags: Sendbcst-pkt-to-re
Addresses, Flags: Dest-route-down Is-Preferred Is-Primary
Destination: 1.1.1/24, Local: 1.1.1.1, Broadcast: 1.1.1.255,
Generation: 150

```

## Sample Output

### show interfaces terse

```

user@host> show interfaces terse

```

Interface	Admin	Link	Proto	Local	Remote
ge-0/0/0	up	up			
ge-0/0/0.0	up	up	inet	10.209.4.61/18	
gr-0/0/0	up	up			
ip-0/0/0	up	up			
st0	up	up			
st0.1	up	ready	inet		
ls-0/0/0	up	up			
lt-0/0/0	up	up			
mt-0/0/0	up	up			
pd-0/0/0	up	up			
pe-0/0/0	up	up			
e3-1/0/0	up	up			
t3-2/0/0	up	up			
e1-3/0/0	up	up			
se-4/0/0	up	down			
t1-5/0/0	up	up			
br-6/0/0	up	up			
dc-6/0/0	up	up			
dc-6/0/0.32767	up	up			
bc-6/0/0:1	down	up			
bc-6/0/0:1.0	up	down			
d10	up	up			
d10.0	up	up	inet		
dsc	up	up			
gre	up	up			
ipip	up	up			
lo0	up	up			
lo0.16385	up	up	inet	10.0.0.1 10.0.0.16	--> 0/0 --> 0/0
lsi	up	up			
mtun	up	up			
pimd	up	up			
pime	up	up			
pp0	up	up			

## Sample Output

### show interfaces controller (Channelized E1 IQ with Logical E1)

```

user@host> show interfaces controller ce1-1/2/6

```

Controller	Admin	Link
ce1-1/2/6	up	up
e1-1/2/6	up	up

**show interfaces controller (Channelized E1 IQ with Logical DSO)**

```

user@host> show interfaces controller ce1-1/2/3

Controller                               Admin Link
ce1-1/2/3                               up      up

ds-1/2/3:1                             up      up

ds-1/2/3:2                             up      up

```

**Sample Output****show interfaces descriptions**

```

user@host> show interfaces descriptions
Interface      Admin Link Description
so-1/0/0       up    up    M20-3#1
so-2/0/0       up    up    GSR-12#1
ge-3/0/0       up    up    SMB-OSPF_Area300
so-3/3/0       up    up    GSR-13#1
so-3/3/1       up    up    GSR-13#2
ge-4/0/0       up    up    T320-7#1
ge-5/0/0       up    up    T320-7#2
so-7/1/0       up    up    M160-6#1
ge-8/0/0       up    up    T320-7#3
ge-9/0/0       up    up    T320-7#4
so-10/0/0      up    up    M160-6#2
so-13/0/0      up    up    M20-3#2
so-14/0/0      up    up    GSR-12#2
ge-15/0/0      up    up    SMB-OSPF_Area100
ge-15/0/1      up    up    GSR-13#3

```

**Sample Output****show interfaces destination-class all**

```

user@host> show interfaces destination-class all
Logical interface so-4/0/0.0

      Destination class      Packets      Bytes
                             (packet-per-second) (bits-per-second)
      gold                   0              0
                             (              0) (              0)
      silver                 0              0
                             (              0) (              0)
Logical interface so-0/1/3.0

      Destination class      Packets      Bytes
                             (packet-per-second) (bits-per-second)
      gold                   0              0
                             (              0) (              0)
      silver                 0              0
                             (              0) (              0)

```

**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

```

## Sample Output

### show interfaces far-end-interval coc12-5/2/0

```

user@host> show interfaces far-end-interval coc12-5/2/0
Physical interface: coc12-5/2/0, SNMP ifIndex: 121
05:30-current:
  ES-L: 1, SES-L: 1, UAS-L: 0
05:15-05:30:
  ES-L: 0, SES-L: 0, UAS-L: 0
05:00-05:15:
  ES-L: 0, SES-L: 0, UAS-L: 0
04:45-05:00:

```

```

    ES-L: 0, SES-L: 0, UAS-L: 0
04:30-04:45:
    ES-L: 0, SES-L: 0, UAS-L: 0
04:15-04:30:
    ES-L: 0, SES-L: 0, UAS-L: 0
04:00-04:15:
...

```

#### show interfaces far-end-interval coc1-5/2/1:1

```

user@host> run show interfaces far-end-interval coc1-5/2/1:1
Physical interface: coc1-5/2/1:1, SNMP ifIndex: 342
05:30-current:
    ES-L: 1, SES-L: 1, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
05:15-05:30:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
05:00-05:15:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
04:45-05:00:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
04:30-04:45:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
04:15-04:30:
    ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0, SES-P: 0, UAS-P: 0
04:00-04:15:

```

## Sample Output

#### show interfaces filters

```

user@host> show interfaces filters

```

Interface	Admin	Link	Proto	Input Filter	Output Filter
ge-0/0/0	up	up			
ge-0/0/0.0	up	up	inet		
			iso		
ge-5/0/0	up	up			
ge-5/0/0.0	up	up	any		f-any
			inet		f-inet
			multiservice		
gr-0/3/0	up	up			
ip-0/3/0	up	up			
mt-0/3/0	up	up			
pd-0/3/0	up	up			
pe-0/3/0	up	up			
vt-0/3/0	up	up			
at-1/0/0	up	up			
at-1/0/0.0	up	up	inet		
			iso		
at-1/1/0	up	down			
at-1/1/0.0	up	down	inet		
			iso		
....					

## 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 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

```

## Sample Output

### show interfaces interval (Channelized OC12)

```

user@host> show interfaces interval t3-0/3/0:0
Physical interface: t3-0/3/0:0, SNMP ifIndex: 23
17:43-current:
  LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
  SEFS: 0, UAS: 0
17:28-17:43:
  LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,

```

```

SEFS: 0, UAS: 0
17:13-17:28:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
SEFS: 0, UAS: 0
16:58-17:13:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
SEFS: 0, UAS: 0
16:43-16:58:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
...
Interval Total:
LCV: 230, PCV: 1145859, CCV: 455470, LES: 0, PES: 230, PSES: 230,
CES: 230, CSES: 230, SEFS: 230, UAS: 238

```

### show interfaces interval (E3)

```

user@host> show interfaces interval e3-0/3/0
Physical interface: e3-0/3/0, SNMP ifIndex: 23
17:43-current:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
SEFS: 0, UAS: 0
17:28-17:43:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
SEFS: 0, UAS: 0
17:13-17:28:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
SEFS: 0, UAS: 0
16:58-17:13:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
SEFS: 0, UAS: 0
16:43-16:58:
LCV: 0, PCV: 0, CCV: 0, LES: 0, PES: 0, PSES: 0, CES: 0, CSES: 0,
....
Interval Total:
LCV: 230, PCV: 1145859, CCV: 455470, LES: 0, PES: 230, PSES: 230,
CES: 230, CSES: 230, SEFS: 230, UAS: 238

```

### show interfaces interval (SONET/SDH)

```

user@host> show interfaces interval so-0/1/0
Physical interface: so-0/1/0, SNMP ifIndex: 19
20:02-current:
ES-S: 0, SES-S: 0, SEFS-S: 0, ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0,
SES-P: 0, UAS-P: 0
19:47-20:02:
ES-S: 267, SES-S: 267, SEFS-S: 267, ES-L: 267, SES-L: 267, UAS-L: 267,
ES-P: 267, SES-P: 267, UAS-P: 267
19:32-19:47:
ES-S: 56, SES-S: 56, SEFS-S: 56, ES-L: 56, SES-L: 56, UAS-L: 46, ES-P: 56,
SES-P: 56, UAS-P: 46
19:17-19:32:
ES-S: 0, SES-S: 0, SEFS-S: 0, ES-L: 0, SES-L: 0, UAS-L: 0, ES-P: 0,
SES-P: 0, UAS-P: 0
19:02-19:17:
.....

```

## Sample Output

### show interfaces load-balancing

```
user@host> show interfaces load-balancing
Interface State      Last change  Member count
ams0      Up            1d 00:50    2
ams1      Up            00:00:59    2
```

### show interfaces load-balancing detail

```
user@host> show interfaces load-balancing detail
Load-balancing interfaces detail
Interface      : ams0
State          : Up
Last change    : 1d 00:51
Member count   : 2
Members       :
  Interface    Weight  State
  mams-2/0/0   10     Active
  mams-2/1/0   10     Active
```

## Sample Output

### show interfaces mac-database (All MAC Addresses on a Port)

```
user@host> show interfaces mac-database xe-0/3/3
Physical interface: xe-0/3/3, Enabled, Physical link is Up
  Interface index: 372, SNMP ifIndex: 788
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps, Loopback:
None, Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None

Logical interface xe-0/3/3.0 (Index 364) (SNMP ifIndex 829)
  Flags: SNMP-Traps 0x4004000 Encapsulation: ENET2
MAC address      Input frames  Input bytes  Output frames  Output bytes
00:00:00:00:00:00      1           56           0             0
00:00:c0:01:01:02    7023810     323095260    0             0
00:00:c0:01:01:03    7023810     323095260    0             0
00:00:c0:01:01:04    7023810     323095260    0             0
00:00:c0:01:01:05    7023810     323095260    0             0
00:00:c0:01:01:06    7023810     323095260    0             0
00:00:c0:01:01:07    7023810     323095260    0             0
00:00:c0:01:01:08    7023809     323095214    0             0
00:00:c0:01:01:09    7023809     323095214    0             0
00:00:c0:01:01:0a    7023809     323095214    0             0
00:00:c0:01:01:0b    7023809     323095214    0             0
00:00:c8:01:01:02    30424784    1399540064    37448598      1722635508
00:00:c8:01:01:03    30424784    1399540064    37448598      1722635508
00:00:c8:01:01:04    30424716    1399536936    37448523      1722632058
00:00:c8:01:01:05    30424789    1399540294    37448598      1722635508
00:00:c8:01:01:06    30424788    1399540248    37448597      1722635462
00:00:c8:01:01:07    30424783    1399540018    37448597      1722635462
00:00:c8:01:01:08    30424783    1399540018    37448596      1722635416
00:00:c8:01:01:09    8836796     406492616     8836795      406492570
```

```

00:00:c8:01:01:0a      30424712      1399536752      37448521      1722631966
00:00:c8:01:01:0b      30424715      1399536890      37448523      1722632058
Number of MAC addresses : 21

```

### show interfaces mac-database (All MAC Addresses on a Service)

```

user@host> show interfaces mac-database xe-0/3/3
Logical interface xe-0/3/3.0 (Index 364) (SNMP ifIndex 829)
  Flags: SNMP-Traps 0x4004000 Encapsulation: ENET2
  MAC address      Input frames    Input bytes    Output frames    Output bytes
00:00:00:00:00:00          1             56             0                0
00:00:c0:01:01:02      7023810       323095260      0                0
00:00:c0:01:01:03      7023810       323095260      0                0
00:00:c0:01:01:04      7023810       323095260      0                0
00:00:c0:01:01:05      7023810       323095260      0                0
00:00:c0:01:01:06      7023810       323095260      0                0
00:00:c0:01:01:07      7023810       323095260      0                0
00:00:c0:01:01:08      7023809       323095214      0                0
00:00:c0:01:01:09      7023809       323095214      0                0
00:00:c0:01:01:0a      7023809       323095214      0                0
00:00:c0:01:01:0b      7023809       323095214      0                0
00:00:c8:01:01:02      31016568     1426762128     38040381         1749857526
00:00:c8:01:01:03      31016568     1426762128     38040382         1749857572
00:00:c8:01:01:04      31016499     1426758954     38040306         1749854076
00:00:c8:01:01:05      31016573     1426762358     38040381         1749857526
00:00:c8:01:01:06      31016573     1426762358     38040381         1749857526
00:00:c8:01:01:07      31016567     1426762082     38040380         1749857480
00:00:c8:01:01:08      31016567     1426762082     38040379         1749857434
00:00:c8:01:01:09      9428580      433714680      9428580          433714680
00:00:c8:01:01:0a      31016496     1426758816     38040304         1749853984
00:00:c8:01:01:0b      31016498     1426758908     38040307         1749854122

```

### show interfaces mac-database mac-address

```

user@host> show interfaces mac-database xe-0/3/3 mac-address 00:00:c8:01:01:09
Physical interface: xe-0/3/3, Enabled, Physical link is Up
  Interface index: 372, SNMP ifIndex: 788
  Link-level type: Ethernet, MTU: 1514, LAN-PHY mode, Speed: 10Gbps, Loopback:
None, Source filtering: Disabled, Flow control: Enabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None

Logical interface xe-0/3/3.0 (Index 364) (SNMP ifIndex 829)
  Flags: SNMP-Traps 0x4004000 Encapsulation: ENET2
  MAC address: 00:00:c8:01:01:09, Type: Configured,
    Input bytes   : 202324652
    Output bytes  : 202324560
    Input frames  : 4398362
    Output frames : 4398360
  Policer statistics:
  Policer type    Discarded frames    Discarded bytes
Output aggregate      3992386          183649756

```



## Sample Output

### show interfaces mc-ae

```
user@host> show interfaces mc-ae ae0 unit 512
Member Links   : ae0
Local Status   : active
Peer Status    : active
Logical Interface : ae0.512
Core Facing Interface : Label Ethernet Interface
ICL-PL         : Label Ethernet Interface
```

### show interfaces media (SONET/SDH)

The following example displays the output fields unique to the **show interfaces media** command for a SONET interface (with no level of output specified):

```
user@host> show interfaces media so-4/1/2
Physical interface: so-4/1/2, Enabled, Physical link is Up
  Interface index: 168, SNMP ifIndex: 495
  Link-level type: PPP, MTU: 4474, Clocking: Internal, SONET mode, Speed: OC48,
  Loopback: None, FCS: 16, Payload scrambler: Enabled
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps 16384
  Link flags     : Keepalives
  Keepalive settings: Interval 10 seconds, Up-count 1, Down-count 3
  Keepalive: Input: 1783 (00:00:00 ago), Output: 1786 (00:00:08 ago)
  LCP state: Opened
  NCP state: inet: Not-configured, inet6: Not-configured, iso: Not-configured,
  mpls: Not-configured
  CHAP state: Not-configured
  CoS queues    : 8 supported
  Last flapped  : 2005-06-15 12:14:59 PDT (04:31:29 ago)
  Input rate    : 0 bps (0 pps)
  Output rate   : 0 bps (0 pps)
  SONET alarms  : None
  SONET defects : None
  SONET errors:
    BIP-B1: 121, BIP-B2: 916, REI-L: 0, BIP-B3: 137, REI-P: 16747, BIP-BIP2: 0
  Received path trace: routerb so-1/1/2
  Transmitted path trace: routera so-4/1/2
```

## Sample Output

### show interfaces policers

```
user@host> show interfaces policers
Interface      Admin Link Proto Input Policer      Output Policer
ge-0/0/0       up    up
ge-0/0/0.0     up    up    inet
               up    up    iso
gr-0/3/0       up    up
ip-0/3/0       up    up
mt-0/3/0       up    up
pd-0/3/0       up    up
pe-0/3/0       up    up
...
so-2/0/0       up    up
so-2/0/0.0     up    up    inet  so-2/0/0.0-in-policer  so-2/0/0.0-out-policer
```

```

iso
so-2/1/0      up    down
...

```

### show interfaces policers interface-name

```

user@host> show interfaces policers so-2/1/0
Interface      Admin Link Proto Input Policer      Output Policer
so-2/1/0        up    down
so-2/1/0.0      up    down inet  so-2/1/0.0-in-policer so-2/1/0.0-out-policer
                                     iso
                                     inet6

```

## Sample Output

### show interfaces queue

The following truncated example shows the CoS queue sizes for queues 0, 1, and 3. Queue 1 has a queue buffer size (guaranteed allocated memory) of 9192 bytes.

```

user@host> show interfaces queue
Physical interface: ge-0/0/0, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 509
  Forwarding classes: 8 supported, 8 in use
  Egress queues: 8 supported, 8 in use
  Queue: 0, Forwarding classes: class0
    Queued:
      Packets      :                0                0 pps
      Bytes        :                0                0 bps
    Transmitted:
      Packets      :                0                0 pps
      Bytes        :                0                0 bps
      Tail-dropped packets :                0                0 pps
      RL-dropped packets  :                0                0 pps
      RL-dropped bytes   :                0                0 bps
      RED-dropped packets :                0                0 pps
      Low              :                0                0 pps
      Medium-low       :                0                0 pps
      Medium-high      :                0                0 pps
      High             :                0                0 pps
      RED-dropped bytes :                0                0 bps
      Low              :                0                0 bps
      Medium-low       :                0                0 bps
      Medium-high      :                0                0 bps
      High             :                0                0 bps
    Queue Buffer Usage:
      Reserved buffer   :            118750000 bytes
      Queue-depth bytes :
      Current           :                0
    ..
    ..
  Queue: 1, Forwarding classes: class1
    ..
    ..
    Queue Buffer Usage:
      Reserved buffer   :                9192 bytes
      Queue-depth bytes :
      Current           :                0
    ..

```

```

..
Queue: 3, Forwarding classes: class3
  Queued:
..
..
Queue Buffer Usage:
  Reserved buffer      :          6250000 bytes
  Queue-depth bytes    :
  Current              :          0
..
..

```

## Sample Output

### show interfaces redundancy

```

user@host> show interfaces redundancy
Interface State      Last change Primary    Secondary Current status
rsp0      Not present                sp-1/0/0  sp-0/2/0  both down
rsp1      On secondary 1d 23:56   sp-1/2/0  sp-0/3/0  primary down
rsp2      On primary   10:10:27   sp-1/3/0  sp-0/2/0  secondary down
rlsq0     On primary   00:06:24   lsq-0/3/0 lsq-1/0/0 both up

```

### show interfaces redundancy (Aggregated Ethernet)

```

user@host> show interfaces redundancy
Interface State      Last change Primary    Secondary Current status
rlsq0     On secondary 00:56:12   lsq-4/0/0 lsq-3/0/0 both up

ae0
ae1
ae2
ae3
ae4

```

### show interfaces redundancy detail

```

user@host> show interfaces redundancy detail
Interface      : rlsq0
State          : On primary
Last change    : 00:45:47
Primary        : lsq-0/2/0
Secondary      : lsq-1/2/0
Current status : both up
Mode           : hot-standby

Interface      : rlsq0:0
State          : On primary
Last change    : 00:45:46
Primary        : lsq-0/2/0:0
Secondary      : lsq-1/2/0:0
Current status : both up
Mode           : warm-standby

```

## Sample Output

### show interfaces routing brief

```

user@host> show interfaces routing brief

```

Interface	State	Addresses
so-5/0/3.0	Down	ISO enabled
so-5/0/2.0	Up	MPLS enabled
		ISO enabled
		INET 192.168.2.120
		INET enabled
so-5/0/1.0	Up	MPLS enabled
		ISO enabled
		INET 192.168.2.130
		INET enabled
at-1/0/0.3	Up	CCC enabled
at-1/0/0.2	Up	CCC enabled
at-1/0/0.0	Up	ISO enabled
		INET 192.168.90.10
		INET enabled
lo0.0	Up	ISO 47.0005.80ff.f800.0000.0108.0001.1921.6800.5061.00
		ISO enabled
		INET 127.0.0.1
fxp1.0	Up	
fxp0.0	Up	INET 192.168.6.90

### show interfaces routing detail

```

user@host> show interfaces routing detail
so-5/0/3.0
  Index: 15, Refcount: 2, State: Up <Broadcast PointToPoint Multicast> Change:<>

  Metric: 0, Up/down transitions: 0, Full-duplex
  Link layer: HDLC serial line Encapsulation: PPP Bandwidth: 155Mbps
  ISO address (null)
    State: <Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
so-5/0/2.0
  Index: 14, Refcount: 7, State: <Up Broadcast PointToPoint Multicast> Change:<>

  Metric: 0, Up/down transitions: 0, Full-duplex
  Link layer: HDLC serial line Encapsulation: PPP Bandwidth: 155Mbps
  MPLS address (null)
    State: <Up Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4458 bytes
  ISO address (null)
    State: <Up Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
  INET address 192.168.2.120
    State: <Up Broadcast PointToPoint Multicast Localup> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
    Local address: 192.168.2.120
    Destination: 192.168.2.110/32
  INET address (null)
    State: <Up Broadcast PointToPoint Multicast> Change: <>
    Preference: 0 (120 down), Metric: 0, MTU: 4470 bytes
...

```

### Sample Output

#### show interfaces routing-instance all

```

user@host> show interfaces terse routing-instance all
Interface  Admin  Link  Proto  Local  Remote Instance
at-0/0/1   up     up    inet   10.0.0.1/24

```

```

ge-0/0/0.0 up up inet 192.168.4.28/24 sample-a
at-0/1/0.0 up up inet6 fe80::a:0:0:4/64 sample-b
so-0/0/0.0 up up inet 10.0.0.1/32

```

## Sample Output

### show interfaces snmp-index

```

user@host> show interfaces snmp-index 33
Physical interface: so-2/1/1, Enabled, Physical link is Down
Interface index: 149, SNMP ifIndex: 33
Link-level type: PPP, MTU: 4474, Clocking: Internal, SONET mode, Speed: OC48,
Loopback: None, FCS: 16, Payload scrambler: Enabled
Device flags : Present Running Down
Interface flags: Hardware-Down Point-To-Point SNMP-Traps 16384
Link flags : Keepalives
CoS queues : 8 supported
Last flapped : 2005-06-15 11:45:57 PDT (05:38:43 ago)
Input rate : 0 bps (0 pps)
Output rate : 0 bps (0 pps)
SONET alarms : LOL, PLL, LOS
SONET defects : LOL, PLL, LOF, LOS, SEF, AIS-L, AIS-P

```

## Sample Output

### show interfaces source-class all

```

user@host> show interfaces source-class all
Logical interface so-0/1/0.0

      Source class          Packets          Bytes
                        (packet-per-second) (bits-per-second)
      gold                1928095          161959980
      (                   889) (                   597762)
      bronze                0                0
      (                   0) (                   0)
      silver                0                0
      (                   0) (                   0)
Logical interface so-0/1/3.0

      Source class          Packets          Bytes
                        (packet-per-second) (bits-per-second)
      gold                0                0
      (                   0) (                   0)
      bronze                0                0
      (                   0) (                   0)
      silver             116113          9753492
      (                   939) (                   631616)

```

## Sample Output

### show interfaces statistics (Fast Ethernet)

```

user@host> show interfaces fe-1/3/1 statistics
Physical interface: fe-1/3/1, Enabled, Physical link is Up
Interface index: 144, SNMP ifIndex: 1042
Description: ford fe-1/3/1
Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Loopback: Disabled,
Source filtering: Disabled, Flow control: Enabled
Device flags : Present Running
Interface flags: SNMP-Traps Internal: 0x4000

```

```

CoS queues      : 4 supported, 4 maximum usable queues
Current address: 00:90:69:93:04:dc, Hardware address: 00:90:69:93:04:dc
Last flapped   : 2006-04-18 03:08:59 PDT (00:01:24 ago)
Statistics last cleared: Never
Input rate     : 0 bps (0 pps)
Output rate    : 0 bps (0 pps)
Input errors: 0, Output errors: 0
Active alarms  : None
Active defects : None
Logical interface fe-1/3/1.0 (Index 69) (SNMP ifIndex 50)
  Flags: SNMP-Traps Encapsulation: ENET2
  Protocol inet, MTU: 1500
    Flags: Is-Primary, DCU, SCU-in

      Destination class      Packets      Bytes
                             (packet-per-second) (bits-per-second)
      silver1                0              0
      (                      0) (              0)
      silver2                0              0
      (                      0) (              0)
      silver3                0              0
      (                      0) (              0)

  Addresses, Flags: Is-Default Is-Preferred Is-Primary
  Destination: 10.27.245/24, Local: 10.27.245.2,
  Broadcast: 10.27.245.255
  Protocol iso, MTU: 1497
  Flags: Is-Primary

```

## Sample Output

### show interfaces switch-port

```

user@host# show interfaces ge-slot/0/0 switch-port port-number
Port 0, Physical link is Up
  Speed: 100mbps, Auto-negotiation: Enabled
  Statistics:
    Receive      Transmit
    Total bytes  28437086  21792250
    Total packets 409145    88008
    Unicast packets 9987      83817
    Multicast packets 145002    0
    Broadcast packets 254156    4191
    Multiple collisions 23        10
    FIFO/CRC/Align errors 0         0
    MAC pause frames 0         0
    Oversized frames 0
    Runt frames 0
    Jabber frames 0
    Fragment frames 0
    Discarded frames 0
  Autonegotiation information:
    Negotiation status: Complete
    Link partner:
      Link mode: Full-duplex, Flow control: None, Remote fault: OK, Link
partner Speed: 100 Mbps
    Local resolution:
      Flow control: None, Remote fault: Link OK

```

## Sample Output

### show interfaces transport pm

```

user@host> show interfaces transport pm all current et-0/1/0
Physical interface: et-0/1/0, SNMP ifIndex 515
14:45-current Elapse time:900 Seconds
Near End Suspect Flag:False Reason:None
PM COUNT THRESHOLD TCA-ENABLED TCA-RAISED

OTU-BBE 0 800 No No
OTU-ES 0 135 No No
OTU-SES 0 90 No No
OTU-UAS 427 90 No No
Far End Suspect Flag:True Reason:Unknown
PM COUNT THRESHOLD TCA-ENABLED TCA-RAISED

OTU-BBE 0 800 No No
OTU-ES 0 135 No No
OTU-SES 0 90 No No
OTU-UAS 0 90 No No
Near End Suspect Flag:False Reason:None
PM COUNT THRESHOLD TCA-ENABLED TCA-RAISED

ODU-BBE 0 800 No No
ODU-ES 0 135 No No
ODU-SES 0 90 No No
ODU-UAS 427 90 No No
Far End Suspect Flag:True Reason:Unknown
PM COUNT THRESHOLD TCA-ENABLED TCA-RAISED

ODU-BBE 0 800 No No
ODU-ES 0 135 No No
ODU-SES 0 90 No No
ODU-UAS 0 90 No No
FEC Suspect Flag:False Reason:None
PM COUNT THRESHOLD TCA-ENABLED TCA-RAISED

FEC-CorrectedErr 2008544300 0 NA NA
FEC-UncorrectedWords 0 0 NA NA
BER Suspect Flag:False Reason:None
PM MIN MAX AVG THRESHOLD TCA-ENABLED
TCA-RAISED
BER 3.6e-5 5.8e-5 3.6e-5 10.0e-3 No
Yes
Physical interface: et-0/1/0, SNMP ifIndex 515
14:45-current
Suspect Flag:True Reason:Object Disabled
PM CURRENT MIN MAX AVG THRESHOLD
TCA-ENABLED TCA-RAISED (MIN)
(MAX) (MIN) (MAX) (MIN) (MAX)
Lane chromatic dispersion 0 0 0 0
0 NA NA NA NA
Lane differential group delay 0 0 0 0
0 NA NA NA NA
q Value 120 120 120 120 0
0 NA NA NA NA
SNR 28 28 29 28 0
0 NA NA NA NA

```

Tx output power(0.01dBm)	-5000	-5000	-5000	-5000	-300
-100 No No No No	No	No	No	No	No
Rx input power(0.01dBm)	-3642	-3665	-3626	-3637	-1800
-500 No No No No	No	No	No	No	No
Module temperature(Celsius)	46	46	46	46	-5
75 No No No No	No	No	No	No	No
Tx laser bias current(0.1mA)	0	0	0	0	0
0 NA NA NA NA	NA	NA	NA	NA	NA
Rx laser bias current(0.1mA)	1270	1270	1270	1270	0
0 NA NA NA NA	NA	NA	NA	NA	NA
Carrier frequency offset(MHz)	-186	-186	-186	-186	-5000
5000 No No No No	No	No	No	No	No

## 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 interfaces diagnostics optics

**Supported Platforms** [SRX Series, vSRX](#)

**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.

**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 412](#)

**Output Fields** [Table 35 on page 409](#) lists the output fields for the show interfaces diagnostics optics command. Output fields are listed in the general order in which they appear.

**Table 35: 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.

Table 35: show interfaces diagnostics optics Output Fields (*continued*)

Field Name	Field Description
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 <b>On</b> or <b>Off</b> .
Laser bias current low alarm	Displays whether the laser bias power setting low alarm is <b>On</b> or <b>Off</b> .
Laser bias current high warning	Displays whether the laser bias power setting high warning is <b>On</b> or <b>Off</b> .
Laser bias current low warning	Displays whether the laser bias power setting low warning is <b>On</b> or <b>Off</b> .
Laser output power high alarm	Displays whether the laser output power high alarm is <b>On</b> or <b>Off</b> .
Laser output power low alarm	Displays whether the laser output power low alarm is <b>On</b> or <b>Off</b> .
Laser output power high warning	Displays whether the laser output power high warning is <b>On</b> or <b>Off</b> .
Laser output power low warning	Displays whether the laser output power low warning is <b>On</b> or <b>Off</b> .
Module temperature high alarm	Displays whether the module temperature high alarm is <b>On</b> or <b>Off</b> .
Module temperature low alarm	Displays whether the module temperature low alarm is <b>On</b> or <b>Off</b> .
Module temperature high warning	Displays whether the module temperature high warning is <b>On</b> or <b>Off</b> .
Module temperature low warning	Displays whether the module temperature low warning is <b>On</b> or <b>Off</b> .
Module voltage high alarm	Displays whether the module voltage high alarm is <b>On</b> or <b>Off</b> .
Module voltage low alarm	Displays whether the module voltage low alarm is <b>On</b> or <b>Off</b> .
Module voltage high warning	Displays whether the module voltage high warning is <b>On</b> or <b>Off</b> .
Module voltage low warning	Displays whether the module voltage low warning is <b>On</b> or <b>Off</b> .

Table 35: show interfaces diagnostics optics Output Fields (*continued*)

Field Name	Field Description
Laser rx power high alarm	Displays whether the receive laser power high alarm is <b>On</b> or <b>Off</b> .
Laser rx power low alarm	Displays whether the receive laser power low alarm is <b>On</b> or <b>Off</b> .
Laser rx power high warning	Displays whether the receive laser power high warning is <b>On</b> or <b>Off</b> .
Laser rx power low warning	Displays whether the receive laser power low warning is <b>On</b> or <b>Off</b> .
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.
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.

Table 35: show interfaces diagnostics optics Output Fields (*continued*)

Field Name	Field Description
<b>Module voltage low alarm threshold</b>	Displays the vendor-specified threshold for the module voltage low alarm.
<b>Module voltage high warning threshold</b>	Displays the vendor-specified threshold for the module voltage high warning.
<b>Module voltage low warning threshold</b>	Displays the vendor-specified threshold for the module voltage low warning.
<b>Laser rx power high alarm threshold</b>	Displays the vendor-specified threshold for the laser rx power high alarm.
<b>Laser rx power low alarm threshold</b>	Displays the vendor-specified threshold for the laser rx power low alarm.
<b>Laser rx power high warning threshold</b>	Displays the vendor-specified threshold for the laser rx power high warning.
<b>Laser rx power low warning threshold</b>	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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [Understanding Interfaces](#)

**List of Sample Output** [show interfaces flow-statistics \(Gigabit Ethernet\) on page 417](#)

**Output Fields** [Table 36 on page 415](#) lists the output fields for the `show interfaces flow-statistics` command. Output fields are listed in the approximate order in which they appear.

Table 36: show interfaces flow-statistics Output Fields

Field Name	Field Description
<b>Traffic statistics</b>	Number of packets and bytes transmitted and received on the physical interface.
<b>Local statistics</b>	Number of packets and bytes transmitted and received on the physical interface.
<b>Transit statistics</b>	Number of packets and bytes transiting the physical interface.
<b>Flow input statistics</b>	Statistics on packets received by flow module.
<b>Flow output statistics</b>	Statistics on packets sent by flow module.
<b>Flow error statistics</b>	Packet drop statistics for the flow module.  For further details, see <a href="#">Table 37 on page 415</a> .

Table 37: Flow Error Statistics (Packet Drop Statistics for the Flow Module)

Error	Error Description
<b>Screen:</b>	
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.
<b>VPN:</b>	
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.
Security association not active	The packet was dropped because an IPsec packet was received for an inactive SA.
<b>NAT:</b>	
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.
<b>Auth:</b>	
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.

Table 37: Flow Error Statistics (Packet Drop Statistics for the Flow Module) (*continued*)

User authentication errors	<p>Packet was dropped because policy requires authentication; however:</p> <ul style="list-style-type: none"> <li>• Only Telnet, FTP, and HTTP traffic can be authenticated.</li> <li>• The corresponding authentication entry could not be found, if web-auth is specified.</li> <li>• The maximum number of authenticated sessions per user was exceeded.</li> </ul>
<b>Flow:</b>	
No one interested in self packets	<p>This counter is incremented for one of the following reasons:</p> <ul style="list-style-type: none"> <li>• 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.</li> <li>• No service is interested in the to-self packet</li> <li>• 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.</li> </ul>
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.
No route present	<p>The packet was dropped because a valid route was not available to forward the packet.</p> <p>For new sessions, the counter is incremented for one of the following reasons:</p> <ul style="list-style-type: none"> <li>• No valid route was found to forward the packet.</li> <li>• A discard or reject route was found.</li> <li>• The route could not be added due to lack of memory.</li> <li>• The reverse path forwarding check failed for an incoming multicast packet.</li> </ul> <p>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:</p> <ul style="list-style-type: none"> <li>• A new route could not be found; either the previous route was removed, or the route was changed to discard or reject.</li> <li>• Multiple packets may concurrently force rerouting to occur, and only one packet can successfully complete the rerouting process. Other packets will be dropped.</li> <li>• 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.</li> </ul>
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	<p>The error counter is incremented for one of the following reasons:</p> <ul style="list-style-type: none"> <li>• Source and/or destination NAT has occurred and policy says to drop the packet.</li> <li>• Policy specifies user authentication, which failed.</li> <li>• Policy was configured to deny this packet.</li> </ul>



Table 37: Flow Error Statistics (Packet Drop Statistics for the Flow Module) (*continued*)

TCP sequence number out of window	A TCP packet with a sequence number failed the TCP sequence number check that was received.
<b>Counters Not Currently in Use</b>	
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 419](#)  
[show interfaces swfab1 on page 419](#)

**Output Fields** [Table 38 on page 419](#) lists the output fields for the **show interfaces <swfab0 | swfab1>** command. Output fields are listed in the approximate order in which they appear.

**Table 38: show interfaces <swfab0 | swfab1> Output Fields**

Field Name	Field Description
<b>fabric-options</b>	The fabric-options hierarchy is configured to be in sync with the fab interfaces.
<b>member-interfaces</b>	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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 161](#)
- [monitor security flow filter on page 329](#)
- [monitor security flow start on page 331](#)
- [clear monitor security flow filter on page 301](#)

**List of Sample Output** [show monitor security flow on page 422](#)

**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 39: show monitor security flow Output Fields**

Field Name	Field Description
Monitor security flow session status	State of the security flow session monitoring: <b>active</b> or <b>inactive</b> .
Monitor security flow trace file	Name of the file for monitoring output.
Monitor security flow filters	<ul style="list-style-type: none"> <li>• <b>conn-tag</b>—Tag that uniquely identifies a session. The session key is expanded to include this tuple.</li> <li>• <b>Destination Address</b>—Address of the destination to be matched.</li> <li>• <b>Destination Port</b>—Name of the destination port to be matched.</li> <li>• <b>Interface Name</b>—Interface name to be matched.</li> <li>• <b>Logical System Name</b>—Logical system name to be matched.</li> <li>• <b>Name</b>—Name of the security flow filter.</li> <li>• <b>Protocol</b>—Name of the protocol to be matched.</li> <li>• <b>Source Address</b>—Address of the source to be matched.</li> <li>• <b>Source Port</b>—Name of the source port to be matched.</li> <li>• <b>Status</b>—State of the security flow filter: <b>active</b> or <b>inactive</b>.</li> </ul>

## 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

**Supported Platforms** [SRX Series, vSRX](#)

**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 SRX1500, SRX5800, SRX5600, and SRX400 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** [• Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show security flow cp-session on page 424](#)  
[show security flow cp-session summary on page 425](#)  
[show security flow cp-session terse on page 425](#)

**Output Fields** Table 40 on page 424 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 40: 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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 423](#)
- [show security flow cp-session destination-prefix on page 429](#)

**List of Sample Output** [show security flow cp-session destination-port summary on page 427](#)  
[show security flow cp-session destination-port terse on page 427](#)

**Output Fields** [Table 41 on page 426](#) 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 41: show security flow cp-session destination-port 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.

Table 41: show security flow cp-session destination-port Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 423](#)
- [show security flow cp-session destination-port on page 426](#)

**List of Sample Output** [show security flow cp-session destination-prefix summary on page 430](#)  
[show security flow cp-session destination-prefix terse on page 430](#)

**Output Fields** [Table 42 on page 429](#) 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 42: show security flow cp-session destination-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.
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.

Table 42: show security flow cp-session destination-prefix Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 423](#)

**List of Sample Output** [show security flow cp-session family summary on page 433](#)  
[show security flow cp-session family terse on page 433](#)

**Output Fields** [Table 43 on page 432](#) 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 43: 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.
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.



Table 43: show security flow cp-session family Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 423](#)

**List of Sample Output** [show security flow cp-session protocol summary on page 436](#)

[show security flow cp-session protocol terse on page 437](#)

**Output Fields** [Table 44 on page 436](#) 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 44: 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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 423](#)
- [show security flow cp-session source-prefix on page 441](#)

**List of Sample Output** [show security flow cp-session source-port summary on page 439](#)  
[show security flow cp-session source-port terse on page 439](#)

**Output Fields** [Table 45 on page 438](#) 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 45: 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.
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.

Table 45: show security flow cp-session source-port Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 423](#)
  - [show security flow cp-session source-port on page 438](#)

**List of Sample Output** [show security flow cp-session source-prefix summary on page 442](#)  
[show security flow cp-session source-prefix terse on page 442](#)

**Output Fields** [Table 46 on page 441](#) 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 46: 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.
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.

Table 46: show security flow cp-session source-prefix Output Fields (*continued*)

Field Name	Field Description
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 sessions 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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 456](#)
  - [show security flow gate destination-port on page 462](#)
  - [show security flow gate destination-prefix on page 465](#)
  - [show security flow gate protocol on page 468](#)
  - [show security flow gate summary node on page 471](#)

**List of Sample Output** [show security flow gate on page 445](#)  
[show security flow gate brief on page 446](#)  
[show security flow gate summary on page 447](#)

**Output Fields** [Table 47 on page 445](#) lists the output fields for the **show security flow gate** command. Output fields are listed in the approximate order in which they appear.

Table 47: 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. <ul style="list-style-type: none"> <li>Source address and port</li> <li>Destination address and port</li> </ul>
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

---

**Supported Platforms** [SRX Series](#), [vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow ip-action on page 302](#)
- [clear security flow session destination-port on page 308](#)

**List of Sample Output**

[show security flow ip-action on page 450](#)  
[show security flow ip-action destination-port on page 451](#)  
[show security flow ip-action destination-prefix on page 452](#)  
[show security flow ip-action family inet protocol on page 452](#)  
[show security flow ip-action family inet logical-system all on page 453](#)  
[show security flow ip-action source-prefix on page 454](#)  
[show security flow ip-action summary on page 455](#)  
[show security flow ip-action summary family inet on page 455](#)  
[show security flow ip-action summary family inet6 on page 455](#)

**Output Fields** [Table 48 on page 449](#) 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 48: show security flow ip-action Output Fields**

Field Name	Field Description
Src-Addr	Source address of outbound IP traffic.
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.

Table 48: show security flow ip-action Output Fields (*continued*)

Field Name	Field Description
Action	Configured action type, for example, block, close, and notify.
State	The active mode and passive mode describe the states of the <b>ip-action</b> 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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 444](#)
- [show security flow gate summary node on page 471](#)

**List of Sample Output**

- [show security flow gate brief node 0 on page 457](#)
- [show security flow gate brief node 1 on page 458](#)
- [show security flow gate brief node all on page 458](#)
- [show security flow gate brief node local on page 460](#)
- [show security flow gate brief node primary on page 460](#)

**Output Fields** [Table 49 on page 456](#) 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 49: 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.



Table 49: show security flow gate brief node Output Fields (*continued*)

Field Name	Field Description
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. <ul style="list-style-type: none"> <li>• Source address and port</li> <li>• Destination address and port</li> </ul>
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:

```
Ho1e: 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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 444](#)
- [show security flow gate destination-prefix on page 465](#)

**List of Sample Output** [show security flow gate destination-port brief on page 463](#)  
[show security flow gate destination-port summary on page 464](#)

**Output Fields** [Table 50 on page 462](#) 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 50: 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. <ul style="list-style-type: none"> <li>Source address and port</li> <li>Destination address and port</li> </ul>
Protocol	Application protocol, such as UDP or TCP.

Table 50: show security flow gate destination-port Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 444](#)
  - [show security flow gate destination-port on page 462](#)

**List of Sample Output** [show security flow gate destination-prefix brief on page 466](#)  
[show security flow gate destination-prefix summary on page 467](#)

**Output Fields** [Table 51 on page 465](#) 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 51: 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. <ul style="list-style-type: none"> <li>• Source address and port</li> <li>• Destination address and port</li> </ul>
Protocol	Application protocol, such as UDP or TCP.
Application	Name of the application.

Table 51: show security flow gate destination-prefix Output Fields (*continued*)

Field Name	Field Description
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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 444](#)

**List of Sample Output** [show security flow gate protocol brief on page 469](#)  
[show security flow gate protocol summary on page 470](#)

**Output Fields** Table 52 on page 469 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 52: 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. <ul style="list-style-type: none"> <li>Source address and port</li> <li>Destination address and port</li> </ul>
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

**Supported Platforms** [SRX Series, vSRX](#)

**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 on page 456](#)

**List of Sample Output** [show security flow gate summary node 0 on page 472](#)  
[show security flow gate summary node 1 on page 473](#)  
[show security flow gate summary node all on page 473](#)  
[show security flow gate summary node local on page 474](#)  
[show security flow gate summary node primary on page 475](#)

**Output Fields** [Table 53 on page 471](#) 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 53: 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.

Table 53: show security flow gate summary node Output Fields (*continued*)

Field Name	Field Description
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. <ul style="list-style-type: none"> <li>• Source address and port</li> <li>• Destination address and port</li> </ul>
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 session

---

**Supported Platforms** [SRX Series, vSRX](#)

**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.

**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 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. Because of this, the output displays slightly larger bytes counters for sessions in Express Path mode than 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 advanced-anti-malware option, see the [Sky Advanced Threat Prevention CLI Reference Guide](#).

**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

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session all on page 304](#)

**List of Sample Output**

- [show security flow session on page 480](#)
- [show security flow session brief on page 480](#)
- [show security flow session extensive on page 481](#)
- [show security flow session summary on page 481](#)

**Output Fields** Table 54 on page 478 lists the output fields for the **show security flow session** command. Output fields are listed in the approximate order in which they appear.

**Table 54: 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
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 (SCTP) 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

Table 54: show security flow session Output Fields (*continued*)

Field Name	Field Description	Level of Output
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
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
	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 tsource IP address is translated to an IP address in the address pool.	
Dynamic application	Name of the application.	extensive
Application traffic control rule-set	AppQoS rule set for this session.	extensive

Table 54: show security flow session Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Rule</b>	AppQoS rule for this session.	extensive
<b>Maximum timeout</b>	Maximum session timeout.	extensive
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.	extensive
<b>Session State</b>	Session state.	extensive
<b>Start time</b>	Time when the session was created, offset from the system start time.	extensive
<b>Unicast-sessions</b>	Number of unicast sessions.	Summary
<b>Multicast-sessions</b>	Number of multicast sessions.	Summary
<b>Services-offload-sessions</b>	Number of services-offload sessions.	Summary
<b>Failed-sessions</b>	Number of failed sessions.	Summary
<b>Sessions-in-use</b>	Number of sessions in use. <ul style="list-style-type: none"> <li>Valid sessions</li> <li>Pending sessions</li> <li>Invalidated sessions</li> <li>Sessions in other states</li> </ul>	Summary
<b>Maximum-sessions</b>	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 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 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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [show security flow session](#)

**List of Sample Output**

[show security flow session brief node 0 on page 484](#)  
[show security flow session brief node 1 on page 484](#)  
[show security flow session brief node all on page 485](#)  
[show security flow session brief node local on page 485](#)  
[show security flow session brief node primary on page 486](#)

**Output Fields** [Table 55 on page 483](#) 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 55: 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.

Table 55: show security flow session brief node Output Fields (*continued*)

Field Name	Field Description
<b>Timeout</b>	Idle timeout after which the session expires.
<b>In</b>	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>CP Session ID</b>	Number that identifies the central point session. Use this ID to get more information about the central point session.
<b>Total sessions</b>	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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session destination-port on page 308](#)

**List of Sample Output**

[show security flow session destination-port 23 on page 488](#)  
[show security flow session destination-port 23 brief on page 489](#)  
[show security flow session destination-port 23 extensive on page 489](#)  
[show security flow session destination-port 23 summary on page 490](#)

**Output Fields** [Table 56 on page 487](#) 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 56: 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).

Table 56: show security flow session destination-port Output Fields (*continued*)

Field Name	Field Description
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Total sessions</b>	Total number of sessions.
<b>Status</b>	Session status.
<b>Flag</b>	Internal flag depicting the state of the session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Application</b>	Name of the application.
<b>Maximum timeout</b>	Maximum session timeout.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Session State</b>	Session state.
<b>Start time</b>	Time when the session was created, offset from the system start time.
<b>Unicast-sessions</b>	Number of unicast sessions.
<b>Multicast-sessions</b>	Number of multicast sessions.
<b>Failed-sessions</b>	Number of failed sessions.
<b>Sessions-in-use</b>	Number of sessions in use. <ul style="list-style-type: none"> <li>• Valid sessions</li> <li>• Pending sessions</li> <li>• Invalidated sessions</li> <li>• Sessions in other states</li> </ul>
<b>Maximum-sessions</b>	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

**Supported Platforms** [SRX Series, vSRX](#)

**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**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [clear security flow session destination-port on page 308](#)

**List of Sample Output** [show security flow session destination-prefix 60/8 on page 493](#)  
[show security flow session destination-prefix 60/8 brief on page 493](#)  
[show security flow session destination-prefix 60/8 extensive on page 493](#)  
[show security flow session destination-prefix 60/8 summary on page 494](#)  
[show security flow session destination-prefix 10::10 on page 494](#)

**Output Fields** [Table 57 on page 491](#) 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 57: 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.

Table 57: show security flow session destination-prefix Output Fields (*continued*)

Field Name	Field Description
<b>In</b>	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Total sessions</b>	Total number of sessions.
<b>Status</b>	Session status.
<b>Flag</b>	Internal flag depicting the state of the session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Application</b>	Name of the application.
<b>Maximum timeout</b>	Maximum session timeout.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Session State</b>	Session state.
<b>Start time</b>	Time when the session was created, offset from the system start time.
<b>Unicast-sessions</b>	Number of unicast sessions.
<b>Multicast-sessions</b>	Number of multicast sessions.
<b>Failed-sessions</b>	Number of failed sessions.
<b>Sessions-in-use</b>	Number of sessions in use. <ul style="list-style-type: none"> <li>• Valid sessions</li> <li>• Pending sessions</li> <li>• Invalidated sessions</li> <li>• Sessions in other states</li> </ul>
<b>Maximum-sessions</b>	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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [show security flow session on page 476](#)

**List of Sample Output** [show security flow session extensive node 0 on page 496](#)  
[show security flow session extensive node 1 on page 497](#)  
[show security flow session extensive node all on page 498](#)  
[show security flow session extensive node local on page 499](#)  
[show security flow session extensive node primary on page 500](#)

**Output Fields** [Table 58 on page 495](#) 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 58: 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.

Table 58: show security flow session extensive node Output Fields (*continued*)

Field Name	Field Description
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: 0x80000042/0x80000000/0x110103
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

**Supported Platforms** [SRX Series, vSRX](#)

**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**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [clear security flow session family on page 310](#)

**List of Sample Output**

[show security flow session family inet on page 502](#)  
[show security flow session family inet brief on page 503](#)  
[show security flow session family inet extensive on page 503](#)  
[show security flow session family inet summary on page 505](#)

**Output Fields** [Table 59 on page 501](#) 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 59: 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).

Table 59: show security flow session family Output Fields (*continued*)

Field Name	Field Description
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Total sessions</b>	Total number of sessions.
<b>Status</b>	Session status.
<b>Flag</b>	Internal flag depicting the state of the session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Application</b>	Name of the application.
<b>Maximum timeout</b>	Maximum session timeout.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Session State</b>	Session state.
<b>Start time</b>	Time when the session was created, offset from the system start time.
<b>Unicast-sessions</b>	Number of unicast sessions.
<b>Multicast-sessions</b>	Number of multicast sessions.
<b>Failed-sessions</b>	Number of failed sessions.
<b>Sessions-in-use</b>	Number of sessions in use. <ul style="list-style-type: none"> <li>• Valid sessions</li> <li>• Pending sessions</li> <li>• Invalidated sessions</li> <li>• Sessions in other states</li> </ul>
<b>Maximum-sessions</b>	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: 0xffffb0006, 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: 0xffffb0006, 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: 0xffffb0006, 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

**Supported Platforms** [SRX Series, vSRX](#)

**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**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [clear security flow session interface on page 311](#)

**List of Sample Output** [show security flow session interface ge-0/0/2.0 on page 507](#)  
[show security flow session interface ge-0/0/2.0 brief on page 508](#)  
[show security flow session interface ge-0/0/2.0 extensive on page 508](#)  
[show security flow session interface ge-7/1/1.0 summary on page 509](#)

**Output Fields** [Table 60 on page 506](#) 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 60: 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.
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).

Table 60: show security flow session interface Output Fields (*continued*)

Field Name	Field Description
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Total sessions</b>	Total number of sessions.
<b>Status</b>	Session status.
<b>Flag</b>	Internal flag depicting the state of the session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Application</b>	Name of the application.
<b>Maximum timeout</b>	Maximum session timeout.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Session State</b>	Session state.
<b>Start time</b>	Time when the session was created, offset from the system start time.
<b>Unicast-sessions</b>	Number of unicast sessions.
<b>Multicast-sessions</b>	Number of multicast sessions.
<b>Failed-sessions</b>	Number of failed sessions.
<b>Sessions-in-use</b>	Number of sessions in use. <ul style="list-style-type: none"> <li>• Valid sessions</li> <li>• Pending sessions</li> <li>• Invalidated sessions</li> <li>• Sessions in other states</li> </ul>
<b>Maximum-sessions</b>	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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [show security flow session on page 476](#)

**List of Sample Output**

[show security flow session nat brief on page 511](#)  
[show security flow session nat extensive on page 511](#)  
[show security flow session nat summary on page 512](#)

**Output Fields** [Table 61 on page 510](#) 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 61: 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.
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.

Table 61: show security flow session nat Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [clear security flow session protocol on page 313](#)

**List of Sample Output** [show security flow session policy-id 4 on page 515](#)  
[show security flow session policy-id 4 extensive on page 515](#)

**Output Fields** [Table 62 on page 513](#) 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 62: 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.
Timeout	Idle timeout after which the session expires.

Table 62: show security flow session policy-id Output Fields (*continued*)

Field Name	Field Description
<b>In</b>	<p>For the input flow:</p> <ul style="list-style-type: none"> <li>• Source and destination addresses and protocol tuple for the input flow.</li> <li>• <b>Interface</b>: Input flow interface.</li> <li>• <b>Session token</b>: Internal token derived from the virtual routing instance.</li> <li>• <b>Flag</b>: Internal debugging flags.</li> <li>• <b>Route</b>: Internal next hop of the route to be used by the flow.</li> <li>• <b>Gateway</b>: Next-hop gateway of the flow.</li> <li>• <b>Tunnel</b>: If the flow is going into a tunnel, the tunnel ID. Otherwise, 0 (zero).</li> <li>• <b>Port Sequence, FIN sequence, FIN state, Cookie</b>: Internal TCP state tracking information.</li> </ul>
<b>Out</b>	<p>For the reverse flow:</p> <ul style="list-style-type: none"> <li>• Source and destination addresses, and protocol tuple for the reverse flow.</li> <li>• <b>Interface</b>: Reverse flow interface.</li> <li>• <b>Session token</b>: Internal token derived from the virtual routing instance.</li> <li>• <b>Flag</b>: Internal debugging flags.</li> <li>• <b>Route</b>: Internal next hop of the route to be used by the flow.</li> <li>• <b>Gateway</b>: Next-hop gateway of the flow.</li> <li>• <b>Tunnel</b>: If the flow is going into a tunnel, the tunnel ID. Otherwise, 0 (zero).</li> <li>• <b>Port Sequence, FIN sequence, FIN state, Cookie</b>: Internal TCP state tracking information.</li> </ul>
<b>Total sessions</b>	Total number of sessions.
<b>Status</b>	Session status.
<b>Flag</b>	Internal flag depicting the state of the session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Dynamic application</b>	Name of the application.
<b>Maximum timeout</b>	Maximum session timeout.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Session State</b>	Session state.
<b>Start time</b>	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

---

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session protocol on page 313](#)

**List of Sample Output** [show security flow session protocol icmp on page 518](#)  
[show security flow session protocol icmp brief on page 518](#)  
[show security flow session protocol icmp extensive on page 519](#)  
[show security flow session protocol icmp summary on page 519](#)

**Output Fields** Table 63 on page 517 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 63: show security flow session protocol Output Fields**

Field Name	Field Description
<b>Session ID</b>	Number that identifies the session. You can use this ID to get additional information about the session.
<b>Policy name</b>	Policy that permitted the traffic.
<b>Timeout</b>	Idle timeout after which the session expires.
<b>In</b>	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Total sessions</b>	Total number of sessions.
<b>Status</b>	Session status.
<b>Flag</b>	Internal flag depicting the state of the session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Application</b>	Name of the application.
<b>Maximum timeout</b>	Maximum session timeout.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Session State</b>	Session state.
<b>Start time</b>	Time when the session was created, offset from the system start time.
<b>Unicast-sessions</b>	Number of unicast sessions.
<b>Multicast-sessions</b>	Number of multicast sessions.
<b>Failed-sessions</b>	Number of failed sessions.

Table 63: show security flow session protocol Output Fields (*continued*)

Field Name	Field Description
<b>Sessions-in-use</b>	Number of sessions in use. <ul style="list-style-type: none"> <li>Valid sessions</li> <li>Pending sessions</li> <li>Invalidated sessions</li> <li>Sessions in other states</li> </ul>
<b>Maximum-sessions</b>	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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session resource-manager on page 315](#)

**List of Sample Output** [show security flow session resource-manager on page 522](#)  
[show security flow session resource-manager brief on page 523](#)  
[show security flow session resource-manager extensive on page 523](#)  
[show security flow session resource-manager summary on page 524](#)

**Output Fields** [Table 64 on page 521](#) 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 64: 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.
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).

Table 64: show security flow session resource-manager Output Fields (*continued*)

Field Name	Field Description
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets and bytes).
<b>Total sessions</b>	Total number of sessions.
<b>Status</b>	Session status.
<b>Flag</b>	Internal flag depicting the state of the session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Application</b>	Name of the application.
<b>Maximum timeout</b>	Maximum session timeout.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Session State</b>	Session state.
<b>Start time</b>	Time when the session was created, offset from the system start time.
<b>Valid sessions</b>	Number of valid sessions.
<b>Pending sessions</b>	Number of pending sessions.
<b>Invalidated sessions</b>	Number of invalidated sessions.
<b>Sessions in other states</b>	Number of sessions in other states.
<b>CP Session ID</b>	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

**Supported Platforms** [SRX Series, vSRX](#)

**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 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 SRX 5800, SRX 5600, SRX 5400 devices and vSRX.

**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.

- **brief | extensive | summary**—Display the specified level of output.

**Required Privilege Level**

view

**Related Documentation**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session services-offload on page 317](#)

**List of Sample Output**

[show security flow session services-offload on page 527](#)  
[show security flow session services-offload brief on page 527](#)  
[show security flow session services-offload extensive on page 528](#)  
[show security flow session services-offload summary on page 528](#)

**Output Fields**

[Table 65 on page 526](#) 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 65: show security flow session services-offload Output Fields**

Field Name	Field Description
<b>Session ID</b>	Number that identifies the services-offload session. Use this ID to get more information about the session.
<b>Policy name</b>	Policy that permits the services-offload traffic.
<b>Timeout</b>	Idle timeout period after which the services-offload session expires.
<b>In</b>	Incoming flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets, and bytes).
<b>Out</b>	Reverse flow (source and destination IP addresses, application protocol, interface, session token, route, gateway, tunnel, port sequence, FIN sequence, FIN state, packets, and bytes).
<b>Total sessions</b>	Total number of services-offload sessions.
<b>Status</b>	Services-offload session status.
<b>Flag</b>	Internal flag depicting the state of the services-offload session, used for debugging purposes.
<b>Policy name</b>	Name and ID of the policy that the first packet of the services-offload session matched.
<b>Source NAT pool</b>	The name of the source pool where NAT is used.
<b>Application</b>	Name of the application.

Table 65: show security flow session services-offload Output Fields (*continued*)

Field Name	Field Description
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.
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 session-identifier

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session session-identifier on page 320](#)

**List of Sample Output** [show security flow session session-identifier 420000002 on page 533](#)  
[show security flow session session-identifier 2218 on page 533](#)  
[show security flow session session-identifier 33 on page 534](#)

**Output Fields** [Table 66 on page 530](#) 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 66: show security flow session session-identifier Output Fields**

Field Name	Field Description
<b>Session ID</b>	Number that identifies the session. This number allows you to get additional information about the session.
<b>Flags</b>	Internal flag depicting the state of the session, used for debugging purposes. The three available flags are: <ul style="list-style-type: none"> <li>• flag</li> <li>• natflag</li> <li>• natflag2</li> </ul>
<b>Policy name</b>	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.

Table 66: show security flow session session-identifier Output Fields (*continued*)

Field Name	Field Description
<b>Source NAT pool</b>	<p>The name of the source pool where NAT is used.</p> <p>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.</p> <p>Source NAT is used to allow hosts with private IP addresses to access a public network.</p>
<b>Dynamic application</b>	<p>Name of the dynamic application.</p> <p>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.</p> <p>Traffic with an application ID of junos:UNKNOWN matches a dynamic application of jjunos:UNKNOWN. If there is no such rule defined, the default rule is applied.</p> <p>The term junos:UNKNOWN is a reserved keyword.</p>
<b>Encryption</b>	Type of encryption, if the application traffic is encrypted.
<b>Application traffic control rule-set</b>	Name of the application traffic control rule set.
<b>Maximum timeout</b>	Maximum session timeout, in seconds.
<b>Session state</b>	Session state: active, valid, pending, invalidated, closed, unknown, other.
<b>Start time</b>	Time, in seconds, when the session was created, offset from the system start time.
<b>In</b>	<p>For the input flow:</p> <ul style="list-style-type: none"> <li>• Source and destination addresses, ports, and protocol tuple for the input flow.</li> <li>• <b>Interface</b>: Input flow interface.</li> <li>• <b>Session token</b>: Internal token derived from the virtual routing instance.</li> <li>• <b>Flag</b>: Internal debugging flags.</li> <li>• <b>Route</b>: Internal next hop of the route to be used by the flow.</li> <li>• <b>Gateway</b>: Next-hop gateway of the flow.</li> <li>• <b>Tunnel</b>: 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 <a href="#">Table 67 on page 532</a> for tunnel type identification.</li> <li>• <b>Port Sequence, FIN sequence, FIN state, Cookie</b>: Internal TCP state tracking information.</li> <li>• <b>Pkts, Bytes, CP Session ID</b>: Packets and bytes matched on the wing, and the associated CP session ID of the wing.</li> <li>• <b>Conn tag</b>: Session connection tag for GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.</li> </ul>

Table 66: show security flow session session-identifier Output Fields (*continued*)

Field Name	Field Description
<b>Out</b>	<p>For the reverse flow:</p> <ul style="list-style-type: none"> <li>Source and destination IP addresses, and application protocol for the reverse flow.</li> <li><b>Interface:</b> Reverse flow interface.</li> <li><b>Session token:</b> Internal token derived from the virtual routing instance.</li> <li><b>Flag:</b> Internal debugging flags.</li> <li><b>Route:</b> Internal next hop of the route to be used by the flow.</li> <li><b>Gateway:</b> Next-hop gateway of the flow.</li> <li><b>Tunnel:</b> Tunnel type identifier. Used for internal debugging.</li> </ul> <p>If the flow is going into a tunnel, species the decimal format of the tunnel ID, plus the tunnel type. Otherwise, 0 (zero). See <a href="#">Table 67 on page 532</a> for tunnel type identification.</p> <ul style="list-style-type: none"> <li><b>Port Sequence, FIN sequence, FIN state, Cookie:</b> Internal TCP state tracking information.</li> <li><b>Pkts, Bytes, CP Session ID:</b> Packets and bytes matched on the wing, and the associated CP session ID of the wing.</li> <li><b>Conn tag:</b> Session connection tag for GRPS tunneling protocol, user plane (GTP-U) flow sessions and Stream Control Transmission Protocol (SCTP) flow sessions.</li> </ul>
<b>Status</b>	<p>Session status:</p> <ul style="list-style-type: none"> <li>Auth (NAT flag with NAT_AUTH)</li> <li>Transparent (NAT flag with NAT_TRANSPARENT)</li> <li>Expired (NAT flag with NAT_INVALID)</li> <li>Normal (no flag)</li> </ul>
<b>Virtual system</b>	Virtual system to which the session belongs. Used only for logical systems. If the sessions are for a specified logical system, the logical system name is not displayed.
<b>Application</b>	Application match for applying the rule.
<b>Rule</b>	Name of the application traffic control rule.
<b>Current timeout</b>	Remaining time for the session unless traffic exists in the session.
<b>Duration</b>	Length of time, in seconds, for which the session is active.
<b>Client</b>	Name of the ALG, if there is resource manager.
<b>Group</b>	Group identification number, if there is resource manager.
<b>Resource</b>	Resource identification number, if there is resource manager.

Table 67: Tunnel Type Identification

Binary (first 3 bits)	Hexadecimal (the first 4 bits)	Tunnel Type
0x000	0x0	no tunnel

Table 67: Tunnel Type Identification (*continued*)

Binary (first 3 bits)	Hexadecimal (the first 4 bits)	Tunnel Type
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 420000002

```

root> show security flow session session-identifier 420000002
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: 1586
Session State: Valid
Start time: 165, Duration: 220
  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

```

## Sample Output

### show security flow session session-identifier 2218

```

user@host> show security flow session session-identifier 2218
Flow Sessions on FPC4 PIC1:

Session ID: 2218, Status: Normal,

```

```
Flags: 0x80000040, 0xffffffff, 0xffffffff
Virtual system: Root VSYS(I), Policy name: foo/4
Maximum timeout: 60, Current timeout: 60
Start time: 0, Duration: 0
Client: MGCP ALG, Group: 2047, Resource: 8188
  In: 12.0.102.26/28072 --> 11.0.101.236/23252;udp,
    Interface: ge-0/0/2.0,
    Session token: 0xa, Flag: 0x8094740
    Route: 0xb0010, Gateway: 12.0.102.26, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0, Cookie: 0,
  Out: 11.0.101.236/23252 --> 12.0.102.26/28072;udp,
    Interface: ge-0/0/1.0,
    Session token: 0x8, Flag: 0x8094740
    Route: 0xa0010, Gateway: 11.0.101.236, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0, Cookie: 0,
Total sessions: 1
```

## Sample Output

### show security flow session session-identifier 33

```
user@host> show security flow session session-identifier 33
Flow Sessions on FPC4 PIC1:

Session ID: 33, Status: Normal,
Flags: 0x80000040, 0xffffffff, 0xffffffff
Virtual system: Root VSYS(I), Policy name: default-policy/2
Application: junos-ftp/1
Maximum timeout: 1800, Current timeout: 1492
Start time: 31128, Duration: 121
  In: 10.10.10.1/2851 --> 192.168.0.2/21;tcp,
    Interface: tl-1/0/0.0,
    Session token: 0x6, Flag: 0x80a15e0
    Route: 0x60010, Gateway: 10.10.10.0, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0, Cookie: 0,
  Out: 192.168.0.2/21 --> 10.10.10.1/2851;tcp,
    Interface: ge-0/0/1.0,
    Session token: 0x6, Flag: 0x80a15e0
    Route: 0x90010, Gateway: 192.168.0.2, Tunnel: 0
    Port sequence: 0, FIN sequence: 0,
    FIN state: 0, Cookie: 0,
Total sessions: 1
```

## show security flow session source-port

**Supported Platforms** [SRX Series, vSRX](#)

**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**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [clear security flow session source-port on page 322](#)

**List of Sample Output** [show security flow session source-port 15198 on page 536](#)  
[show security flow session source-port 15198 brief on page 537](#)  
[show security flow session source-port 15198 extensive on page 537](#)  
[show security flow session source-port 15198 summary on page 538](#)

**Output Fields** [Table 68 on page 535](#) 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 68: 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.

Table 68: show security flow session source-port 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 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.
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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session source-prefix on page 324](#)

**List of Sample Output** [show security flow session source-prefix 200.0.0.10 on page 540](#)  
[show security flow session source-prefix 200.0.0.10 brief on page 541](#)  
[show security flow session source-prefix 200.0.0.10 extensive on page 541](#)  
[show security flow session source-prefix 200.0.0.10 summary on page 542](#)

**Output Fields** [Table 69 on page 539](#) 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 69: 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.

Table 69: show security flow session source-prefix 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 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.
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

**Supported Platforms** [SRX Series, vSRX](#)

**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**
- [Juniper Networks Devices Processing Overview on page 3](#)
  - [clear security flow session all on page 304](#)

**List of Sample Output** [show security flow session summary family inet on page 544](#)  
[show security flow session summary family inet6 on page 544](#)

**Output Fields** [Table 70 on page 543](#) 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 70: show security flow session summary Output Fields**

Field Name	Field Description
Valid sessions	Count of valid sessions.
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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [show security flow session on page 476](#)

**List of Sample Output**

[show security flow session summary node 0 on page 546](#)  
[show security flow session summary node 1 on page 547](#)  
[show security flow session summary node all on page 547](#)  
[show security flow session summary node local on page 549](#)  
[show security flow session summary node primary on page 549](#)

**Output Fields** [Table 71 on page 545](#) 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 71: show security flow session summary node Output Fields**

Field Name	Field Description
Unicast-sessions	Number of unicast sessions.

Table 71: show security flow session summary node Output Fields (*continued*)

Field Name	Field Description
<b>Multicast-sessions</b>	Number of multicast sessions.
<b>Failed-sessions</b>	Number of failed sessions.
<b>Sessions-in-use</b>	Number of sessions in use. <ul style="list-style-type: none"> <li>• Valid sessions</li> <li>• Pending sessions</li> <li>• Invalidated sessions</li> <li>• Sessions in other states</li> </ul>
<b>Maximum-sessions</b>	Number of maximum sessions.

## 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

**Supported Platforms** [SRX Series](#), [vSRX](#)

**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.

**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.

**Required Privilege Level** view

**Related Documentation**

- [Juniper Networks Devices Processing Overview on page 3](#)
- [clear security flow session services-offload on page 317](#)

**List of Sample Output**

- [show security flow session summary services-offload on page 552](#)
- [show security flow session summary services-offload application on page 553](#)
- [show security flow session summary services-offload destination-port on page 554](#)

**Output Fields** Table 72 on page 552 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 72: 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: <ul style="list-style-type: none"> <li>• Valid</li> <li>• Pending</li> <li>• Invalidated</li> <li>• Sessions in other states</li> </ul>
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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show security flow session tunnel on page 557](#)  
[show security flow session tunnel brief on page 557](#)  
[show security flow session tunnel extensive on page 558](#)  
[show security flow session tunnel summary extensive \(with fragmentation counters output\) on page 560](#)  
[show security flow session tunnel summary on page 561](#)  
[show security flow session tunnel summary \(with fragmentation counters output\) on page 561](#)

**Output Fields** [Table 73 on page 555](#) 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 73: 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.

Table 73: show security flow session tunnel Output Fields (*continued*)

Field Name	Field Description
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.
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: <i>number</i> , Generated: <i>number</i>	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: <i>number</i> , Tx: <i>number</i> , Generated: <i>number</i>	For tunnels with IPv4 fragments, the number of fragments associated with the tunnel that were received, transmitted, and generated.

Table 73: show security flow session tunnel Output Fields (*continued*)

Field Name	Field Description
Inner IPv6 frag Rx: <i>number</i> , Tx: <i>number</i> , Generated: <i>number</i>	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
  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
  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: 900000000

  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

---

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax** **show security flow statistics**  
<node (*node-id* | all | local | primary) >

**Release Information** Command introduced in Junos OS Release 10.2. Fragmentation counters options introduced in Junos OS Release 15.1X49-90.

**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, it shows the number of active sessions on the SPU, the number of packets processed and forwarded, number of packets dropped, the number of packet fragments received in a flow on the SPU, the number of pre-fragmented packets generated, and the number of post-fragmented packets generated.

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 than 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**—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** • [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show security flow statistics on page 566](#)

**Output Fields** [Table 74 on page 565](#) lists the output fields for the **show security flow statistics** command. Output fields are listed in the approximate order in which they appear.

**Table 74: show security flow statistics Output Fields**

Field Name	Field Description
Current sessions	Number of active sessions on the SPU.

Table 74: show security flow statistics Output Fields (*continued*)

Field Name	Field Description
Packets forwarded	Number of packets received in a security flow of a specific SPU. The packets are processed and forwarded on that SPU.
Packets dropped	<p>Number of packets dropped in a flow on a specific SPU.</p> <p>The packets are received in the flow. However, during processing, the system discovered sanity check errors, security violations, or other conditions that caused the packet to be dropped.</p> <p>See the description for some of the conditions and events that can cause a packet to be dropped.</p>
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:
```

```
-----
Flow Statistics of FPC0 PIC1:
  Current sessions: 0
  Packets forwarded: 71186468
  Packets dropped: 2803
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0
```

```
Flow Statistics of FPC0 PIC2:
  Current sessions: 0
  Packets forwarded: 71162279
  Packets dropped: 2390
  Fragment packets: 0
  Pre fragments generated: 0
  Post fragments generated: 0
```

```
Flow Statistics of FPC0 PIC3:
```

Current sessions: 0  
Packets forwarded: 71147384  
Packets dropped: 2076  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC4 PIC0:  
Current sessions: 0  
Packets forwarded: 71190678  
Packets dropped: 2435  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC4 PIC1:  
Current sessions: 0  
Packets forwarded: 71106410  
Packets dropped: 2209  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC4 PIC2:  
Current sessions: 0  
Packets forwarded: 71254091  
Packets dropped: 2907  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC4 PIC3:  
Current sessions: 0  
Packets forwarded: 71253962  
Packets dropped: 1908  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC5 PIC0:  
Current sessions: 0  
Packets forwarded: 71092466  
Packets dropped: 2700  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC5 PIC1:  
Current sessions: 0  
Packets forwarded: 71385485  
Packets dropped: 3374  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC5 PIC2:  
Current sessions: 0  
Packets forwarded: 71044650  
Packets dropped: 2085  
Fragment packets: 0  
Pre fragments generated: 0

Post fragments generated: 0

Flow Statistics of FPC5 PIC3:

Current sessions: 0  
Packets forwarded: 71262479  
Packets dropped: 3100  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics Summary:

System total valid sessions: 0  
Packets forwarded: 783086352  
Packets dropped: 27987  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

node1:

-----  
Flow Statistics of FPC0 PIC1:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC0 PIC2:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC0 PIC3:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC4 PIC0:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

Flow Statistics of FPC4 PIC1:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0



## Flow Statistics of FPC4 PIC2:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

## Flow Statistics of FPC4 PIC3:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

## Flow Statistics of FPC5 PIC0:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

## Flow Statistics of FPC5 PIC1:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

## Flow Statistics of FPC5 PIC2:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

## Flow Statistics of FPC5 PIC3:

Current sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

## Flow Statistics Summary:

System total valid sessions: 0  
Packets forwarded: 0  
Packets dropped: 0  
Fragment packets: 0  
Pre fragments generated: 0  
Post fragments generated: 0

## show security flow status

**Supported Platforms** [SRX Series, vSRX](#)

**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.

**Description** Display the flow processing modes and logging status.

**Required Privilege Level** view

**Related Documentation** [• Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show security flow status on page 571](#)  
[show security flow status \(IPsec Performance Acceleration\) on page 571](#)  
[show security flow status \(for hash-based datapath forwarding using SRX5K-MPC3-40G10G \(IOC3\) and SRX5K-MPC3-100G10G \(IOC3\) on page 572](#)

**Output Fields** [Table 75 on page 570](#) lists the output fields for the `show security flow status` command. Output fields are listed in the approximate order in which they appear.

**Table 75: show security flow status Output Fields**

Field Name	Field Description
Flow forwarding mode	Flow processing mode. <ul style="list-style-type: none"> <li>• Inet forwarding mode</li> <li>• Inet6 forwarding mode</li> <li>• MPLS forwarding mode</li> <li>• ISO forwarding mode</li> <li>• Session distribution mode</li> <li>• Enhanced route scaling mode</li> </ul>

Table 75: show security flow status Output Fields (*continued*)

Field Name	Field Description
Flow trace status	Flow logging status. <ul style="list-style-type: none"> <li>Flow tracing status</li> <li>Flow tracing options</li> </ul>
flow session distribution	SPU load distribution mode. <ul style="list-style-type: none"> <li>RR-based</li> <li>Hash-based</li> </ul> GTP-U distribution <ul style="list-style-type: none"> <li>Enabled</li> </ul>
Flow packet ordering	packet-ordering mode. <ul style="list-style-type: none"> <li>Hardware</li> <li>Software</li> </ul>
Flow ipsec performance acceleration	IPsec VPN performance acceleration status.

## Sample Output

### show security flow status

```

root> 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
Enhanced route scaling mode: Enabled (reboot needed to disable)
Flow trace status
Flow tracing status: on
Flow tracing options: all
Flow session distribution
Distribution mode: Hash-based
GTP-U distribution: Enabled
Flow packet ordering
Ordering mode: Software (reboot needed to change to software)

```

### show security flow status (IPsec Performance Acceleration)

```

root> 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
GTP-U distribution: Enabled Flow packet ordering

```

Ordering mode: Software (reboot needed to change to software)  
Flow ipsec performance acceleration: on

**show security flow status (for hash-based datapath forwarding using SRX5K-MPC3-40G10G (IOC3) and SRX5K-MPC3-100G10G (IOC3))**

```
root> show security flow status
node0:
-----
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: Hash-based
  GTP-U distribution: Enabled
Flow ipsec performance acceleration: off
Flow packet ordering
  Ordering mode: Hardware

node1:
-----
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: Hash-based
  GTP-U distribution: Enabled
Flow ipsec performance acceleration: off
Flow packet ordering
  Ordering mode: Hardware
```

## show security forwarding-options mirror-filter

**Supported Platforms** [SRX Series, vSRX](#)

**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.

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 \(Security Forwarding Options\) on page 258](#)
- [clear security forward-options mirror filter on page 326](#)

**List of Sample Output** [show security forward-options mirror-filter on page 574](#)

**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 76: show security forward-options mirror-filter**

Field Name	Field Description
<b>mirror-filter-name</b>	Name of the mirror filter configured on the device.
<b>interface-in</b>	Name of the incoming logical interface to be matched for mirroring.
<b>interface-out</b>	Name of the outgoing logical interface to be matched for mirroring.
<b>protocol</b>	Networking protocol name or number to be matched for mirroring.
<b>source-port</b>	Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) source port number to be matched for mirroring.
<b>source-prefix</b>	Source IP prefix or address to be matched for mirroring.
<b>destination-port</b>	Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) destination port number to be matched for mirroring.
<b>destination-prefix</b>	Destination IP prefix or address to be matched for mirroring.
<b>filter-counters</b>	Number of packets matched for mirroring.
<b>output-counter</b>	Number of packets sent to the packet analyzer.

**Sample Output****show security forward-options mirror-filter**

```
user@host> show security forward-options mirror-filter m10
```

```
node0:
```

```
-----
Security mirror status
```

```
mirror-filter-name: m10
interface-in: reth0.0
interface-out: reth1.0
protocol: 6
source-prefix: 192.0.2.0/24
destination-prefix: 41.1.1.0/24
filter-counters: 143
output-counters: 143
```

```
node1:
```

```
-----
Security mirror status
```

```
mirror-filter-name: m10
interface-in: reth0.0
interface-out: reth1.0
protocol: 6
source-prefix: 11.1.1.0/24
destination-prefix: 41.1.1.0/24
filter-counters: 0
output-counters: 0
```

## show security monitoring

**Supported Platforms** [SRX Series, vSRX](#)

**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 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
```

FPC	PIC	CPU	Mem	Flow session current	Flow session maximum	CP session current	CP session 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 (vSRX)

```
user@host>show security monitoring
```

```
user@host> show security monitoring
```

FPC	PIC	CPU	Mem	Flow session current	Flow session maximum	CP session current	CP session 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:
```

FPC	PIC	CPU	Mem	Flow session current	Flow session maximum	CP session current	CP session maximum
0	0	0	67	0	524288	N/A	N/A

```
node1:
```

FPC	PIC	CPU	Mem	Flow session current	Flow session maximum	CP session current	CP session maximum
0	0	0	67	0	524288	N/A	N/A

## show security policies

---

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax** `show security policies`  
`none`  
`<detail>`  
`policy-name policy-name`  
`<global>`

**Release Information** Command modified in Junos OS Release 9.2. Support for IPv6 addresses added in Junos OS Release 10.2. Support for wildcard addresses added in Junos OS Release 11.1. Support for global policy added in Junos OS Release 11.4. Support for services offloading added in Junos OS Release 11.4. Support for source-identities added in Junos OS Release 12.1. The **Description** output field 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 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 added in Junos OS Release 12.3X48-D20. Output field and description for **source-end-user-profile** option added in Junos OS Release 15.1x49-D70. Output field and description for **dynamic-applications** option added in Junos OS Release 15.1x49-D100.

**Description** Display a summary of all security policies configured on the device. If a particular policy is specified, display information specific to that policy.

- Options**
- **none**—Display basic information about all configured policies.
  - **detail**—(Optional) Display a detailed view of all of the policies configured on the device.
  - **policy-name *policy-name***—(Optional) Display information about a specified policy.
  - **global**—(Optional) Display information about global policies.

**Required Privilege Level** view

- Related Documentation**
- [Security Policies Overview](#)
  - [Understanding Security Policy Rules](#)
  - [Understanding Security Policy Elements](#)

**List of Sample Output** [show security policies on page 582](#)  
[show security policies \(Dynamic Applications\) on page 582](#)  
[show security policies policy-name detail on page 583](#)  
[show security policies \(Services-Offload\) on page 584](#)  
[show security policies \(Device Identity\) on page 584](#)  
[show security policies detail on page 584](#)

[show security policies detail \(TCP Options\) on page 585](#)  
[show security policies policy-name \(Negated Address\) on page 585](#)  
[show security policies policy-name detail \(Negated Address\) on page 586](#)  
[show security policies global on page 586](#)

**Output Fields** Table 77 on page 579 lists the output fields for the **show security policies** command. Output fields are listed in the approximate order in which they appear.

**Table 77: show security policies Output Fields**

Field Name	Field Description
<b>From zone</b>	Name of the source zone.
<b>To zone</b>	Name of the destination zone.
<b>Policy</b>	Name of the applicable policy.
<b>Description</b>	Description of the applicable policy.
<b>State</b>	Status of the policy: <ul style="list-style-type: none"> <li>• <b>enabled:</b> 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.</li> <li>• <b>disabled:</b> The policy cannot be used in the policy lookup process, and therefore it is not available for access control.</li> </ul>
<b>Index</b>	Internal number associated with the policy.
<b>Sequence number</b>	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.
<b>Source addresses</b>	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.
<b>Destination addresses</b>	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.
<b>source-end-user-profile</b>	Name of the device identity profile (referred to as <b>end-user-profile</b> in the CLI) that contains attributes, or characteristics of a device. Specification of the device identity profile in the <b>source-end-user-profile</b> 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.
<b>Source addresses (excluded)</b>	Name of the source address excluded from the policy.
<b>Destination addresses (excluded)</b>	Name of the destination address excluded from the policy.

Table 77: show security policies Output Fields (*continued*)

Field Name	Field Description
Source identities	One or more user roles specified for a policy.
Applications	<p>Name of a preconfigured or custom application whose type the packet matches, as specified at configuration time.</p> <ul style="list-style-type: none"> <li>• <b>IP protocol</b>: The Internet protocol used by the application—for example, TCP, UDP, ICMP.</li> <li>• <b>ALG</b>: If an ALG is explicitly associated with the policy, the name of the ALG is displayed. If <b>application-protocol ignore</b> is configured, <b>ignore</b> is displayed. Otherwise, <b>0</b> is displayed. However, even if this command shows <b>ALG: 0</b>, ALGs might be triggered for packets destined to well-known ports on which ALGs are listening, unless ALGs are explicitly disabled or when <b>application-protocol ignore</b> is not configured for custom applications.</li> <li>• <b>Inactivity timeout</b>: Elapsed time without activity after which the application is terminated.</li> <li>• <b>Source port range</b>: The low-high source port range for the session application.</li> </ul>
Dynamic Applications	Application identification based layer 7 dynamic applications.
Destination Address Translation	<p>Status of the destination address translation traffic:</p> <ul style="list-style-type: none"> <li>• <b>drop translated</b>—Drop the packets with translated destination addresses.</li> <li>• <b>drop untranslated</b>—Drop the packets without translated destination addresses.</li> </ul>
Application Firewall	<p>An application firewall includes the following:</p> <ul style="list-style-type: none"> <li>• <b>Rule-set</b>—Name of the rule set.</li> <li>• <b>Rule</b>—Name of the rule. <ul style="list-style-type: none"> <li>• <b>Dynamic applications</b>—Name of the applications.</li> <li>• <b>Dynamic application groups</b>—Name of the application groups.</li> <li>• <b>Action</b>—The action taken with respect to a packet that matches the application firewall rule set. Actions include the following: <ul style="list-style-type: none"> <li>• <b>permit</b></li> <li>• <b>deny</b></li> </ul> </li> </ul> </li> <li>• <b>Default rule</b>—The default rule applied when the identified application is not specified in any rules of the rule set.</li> </ul>

Table 77: show security policies Output Fields (*continued*)

Field Name	Field Description
Action or Action-type	<ul style="list-style-type: none"> <li>The action taken in regard to a packet that matches the policy's tuples. Actions include the following: <ul style="list-style-type: none"> <li><b>permit</b></li> <li><b>firewall-authentication</b></li> <li><b>tunnel ipsec-vpn <i>vpn-name</i></b></li> <li><b>pair-policy <i>pair-policy-name</i></b></li> <li><b>source-nat pool <i>pool-name</i></b></li> <li><b>pool-set <i>pool-set-name</i></b></li> <li><b>interface</b></li> <li><b>destination-nat <i>name</i></b></li> <li><b>deny</b></li> <li><b>reject</b></li> <li><b>services-offload</b></li> </ul> </li> </ul>
Session log	Session log entry that indicates whether the <b>at-create</b> and <b>at-close</b> 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.
Policy statistics	<ul style="list-style-type: none"> <li><b>Input bytes</b>—The total number of bytes presented for processing by the device. <ul style="list-style-type: none"> <li><b>Initial direction</b>—The number of bytes presented for processing by the device from the initial direction.</li> <li><b>Reply direction</b>—The number of bytes presented for processing by the device from the reply direction.</li> </ul> </li> <li><b>Output bytes</b>—The total number of bytes actually processed by the device. <ul style="list-style-type: none"> <li><b>Initial direction</b>—The number of bytes from the initial direction actually processed by the device.</li> <li><b>Reply direction</b>—The number of bytes from the reply direction actually processed by the device.</li> </ul> </li> <li><b>Input packets</b>—The total number of packets presented for processing by the device. <ul style="list-style-type: none"> <li><b>Initial direction</b>—The number of packets presented for processing by the device from the initial direction.</li> <li><b>Reply direction</b>—The number of packets presented for processing by the device from the reply direction.</li> </ul> </li> <li><b>Output packets</b>—The total number of packets actually processed by the device. <ul style="list-style-type: none"> <li><b>Initial direction</b>—The number of packets actually processed by the device from the initial direction.</li> <li><b>Reply direction</b>—The number of packets actually processed by the device from the reply direction.</li> </ul> </li> <li><b>Session rate</b>—The total number of active and deleted sessions.</li> <li><b>Active sessions</b>—The number of sessions currently present because of access control lookups that used this policy.</li> <li><b>Session deletions</b>—The number of sessions deleted since system startup.</li> <li><b>Policy lookups</b>—The number of times the policy was accessed to check for a match.</li> </ul>

Table 77: show security policies Output Fields (*continued*)

Field Name	Field Description
Per policy TCP Options	Configured syn and sequence checks, and the configured TCP MSS value for the initial direction and /or the reverse direction.

## 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

```

### show security policies policy-name detail

```

user@host> show security policies policy-name p1 detail
Policy: p1, action-type: permit, State: enabled, Index: 4
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

```

### show security policies (Services-Offload)

```
user@host> show security policies
Default policy: deny-all
From zone: trust, To zone: untrust
  Policy: p1, State: enabled, Index: 4, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, services-offload, count
From zone: untrust, To zone: trust
  Policy: p2, State: enabled, Index: 5, Scope Policy: 0, Sequence number: 1
    Source addresses: any
    Destination addresses: any
    Source identities: role1, role2, role4
    Applications: any
    Action: permit, services-offload
```

### 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

```

### show security policies detail (TCP Options)

```

user@host> show security policies policy-name policy1 detail
node0:
-----
Policy: policy1, action-type: permit, State: enabled, Index: 7, Scope Policy: 0
Policy Type: Configured
Sequence number: 2
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: 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
Per policy TCP MSS: initial: 800, reverse: 900

```

### 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: 5, Scope Policy: 0, Sequence number: 1
From zones: zone1, zone2
To zones: zone3, zone4 Source addresses: any
Destination addresses: any
Applications: any
Action: permit
```

## show security policies hit-count

**Supported Platforms** SRX Series, vSRX

**Syntax** show security policies hit-count  
 <ascending | descending>  
 <from-zone *zone-name*>  
 <greater-than *count*>  
 <less-than *count*>  
 <to-zone *zone-name*>

**Release Information** Command introduced in Junos OS Release 12.1.

**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 | descending**—(Optional) Display the number of hits for security policies in ascending or descending order.
  - **from-zone *zone-name***—(Optional) Display the number of hits for security policies associated with the named source zone.
  - **greater-than *count***—(Optional) Display 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) Display security policies for which the number of hits is less than the specified number.  
**Range:** 0 through 4,294,967,295
  - **to-zone *zone-name***—(Optional) Display 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](#)

**List of Sample Output**

- [show security policies hit-count on page 588](#)
- [show security policies hit-count ascending on page 588](#)
- [show security policies hit-count descending greater-than 70 less-than 100 on page 589](#)
- [show security policies hit-count from-zone untrust to-zone trust on page 589](#)

**Output Fields** Table 78 on page 588 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 78: show security policies hit-count Output Fields**

Field Name	Field Description
from-zone	Name of the source zone.
to-zone	Name of the destination zone.
policy	Name of the security policy.
hit-count	Number of hits for each security policy.
Number of policy	Number of security policies for which hit counts are displayed.

## Sample Output

### show security policies hit-count

```
user@host> show security policies hit-count
from-zone  to-zone  policy  hit-count
untrust    vrtrust  u2t1    40
untrust    trust   u2t2    20
untrust    trust   u2t3    80
```

Number of policy: 3

## Sample Output

### show security policies hit-count ascending

```
user@host> show security policies hit-count ascending
from-zone  to-zone  policy  hit-count
untrust    trust   u2t2    20
untrust    vrtrust  u2t1    40
untrust    trust   u2t3    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
from-zone  to-zone  policy  hit-count
untrust    trust    u2t2    100
untrust    vrtrust  u2t1    90
untrust    vrtrust  u2t3    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
from-zone  to-zone  policy  hit-count
untrust    trust    u2t2    20
untrust    trust    u2t3    80
```

Number of policy: 2

## show security resource-manager group active

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output**

[show security resource-manager group active on page 591](#)  
[show security resource-manager group active 2048 on page 591](#)  
[show security resource-manager group active node primary on page 591](#)  
[show security resource-manager group active node all on page 591](#)  
[show security resource-manager group active 1024 node all on page 592](#)

**Output Fields** [Table 79 on page 590](#) 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 79: 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.

Table 79: show security resource-manager group Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show security resource-manager resource active on page 594](#)  
[show security resource-manager resource active 5 on page 594](#)  
[show security resource-manager resource active node local on page 594](#)  
[show security resource-manager resource active node primary on page 595](#)

**Output Fields** [Table 80 on page 593](#) 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 80: show security resource-manager resource Output Fields**

Field Name	Field Description
<b>Total resources</b>	Total number of resources in the system.
<b>active resources</b>	Number of active resources.

Table 80: show security resource-manager resource Output Fields (*continued*)

Field Name	Field Description
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

**Supported Platforms** [SRX Series, vSRX](#)

**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** [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show security resource-manager settings on page 597](#)  
[show security resource-manager settings node primary on page 597](#)  
[show security resource-manager settings node all on page 597](#)

**Output Fields** [Table 81 on page 596](#) 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 81: 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

**Supported Platforms** [SRX Series, vSRX](#)

**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**

- [Juniper Networks Devices Processing Overview on page 3](#)

**List of Sample Output** [show security resource-manager summary on page 598](#)

**Output Fields** [Table 82 on page 598](#) 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 82: 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

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax** show security screen ids-option  
screen-name  
<node ( *node-id* | all | local | primary )>

**Release Information** Command introduced in Junos OS Release 8.5. Support for **node** options added in Junos OS Release 9.0. Support for IPv6 extension header screens added in Junos OS Release 12.1X46-D10. Support for UDP **port scan** added in Junos OS Release 12.1X47-D10.

**Description** Display configuration information about the specified security screen.

- Options**
- **screen-name** —Name of the screen.
  - **node**—(Optional) For chassis cluster configurations, display the configuration status of the security screen 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**
- *ids-option*
  - *Example: Configuring Multiple Screening Options*

**List of Sample Output** [show security screen ids-option jscreen on page 601](#)  
[show security screen ids-option jscreen \(IPv6\) on page 602](#)  
[show security screen ids-option jscreen1 node all on page 602](#)

**Output Fields** [Table 83 on page 599](#) 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 83: 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.

Table 83: show security screen ids-option Output Fields (*continued*)

Field Name	Field Description
ICMP address sweep threshold	Maximum 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.
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	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.



Table 83: show security screen ids-option Output Fields (*continued*)

Field Name	Field Description
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 encaps limit	Enable or disable the IPv6 extension distributed (network) storage tunnel encapsulation limit screen option.
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.

## 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 statistics

---

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax** `show security screen statistics (zone zone-name | interface interface-name)  
<logical-system (logical-system-name | all)>  
<node (node-id | all | local | primary)>  
<root-logical-system>`

**Release Information** Command introduced in Junos OS Release 8.5. **node** options added in Junos OS Release 9.0. **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.

**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**—(Optional) Display screen statistics for configured logical systems.
    - ***logical-system-name***—Display screen statistics for the named logical system.
    - **all**—Display screen statistics for all logical systems, including the master (root) logical system.
  - **node**—(Optional) For chassis cluster configurations, display screen statistics on a specific node.
    - ***node-id***—Identification number of a 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.
  - **root-logical-system**—(Optional) Display screen statistics for the master logical system only.

**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 607](#)

[show security screen statistics zone untrust \(IPv6\) on page 607](#)  
[show security screen statistics interface ge-0/0/3 on page 608](#)  
[show security screen statistics interface ge-0/0/1 \(IPv6\) on page 608](#)  
[show security screen statistics interface ge-0/0/1 node primary on page 609](#)  
[show security screen statistics zone trust logical-system all on page 609](#)

**Output Fields** Table 84 on page 605 lists the output fields for the **show security screen statistics** command. Output fields are listed in the approximate order in which they appear.

**Table 84: 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).
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.

Table 84: show security screen statistics Output Fields (*continued*)

Field Name	Field Description
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.
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.

Table 84: show security screen statistics Output Fields (*continued*)

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 softwires

**Supported Platforms** [SRX5400, SRX5600, SRX5800, vSRX](#)

**Syntax** `show security softwires <software-name software-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 software concentrators and details on concentrators with specified name.

**Options** **software-name software-name**—Display the details of the specified software concentrator.

**logical-system (all | logical-system-name)**—Display software 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**

- [Juniper Networks Devices Processing Overview on page 3](#)

## Sample Output

```
user@host> show security softwires
Software 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 software-name SC-CSSI-1
Name of software: 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
Software Name      SC Address      Status  Number of SI connected
sc_1               3000::1         Connected  1
```

## show security zones

**Supported Platforms** [SRX Series, vSRX](#)

**Syntax** `show security zones <zone-name> <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.

- Options**
- **none**—Display information about all zones.
  - **detail | terse**—(Optional) Display the specified level of output.
  - **zone-name**—(Optional) Display information about the specified zone.

**Required Privilege Level** view

- Related Documentation**
- *Security Zones and Interfaces Overview*
  - *Supported System Services for Host Inbound Traffic*
  - *security-zone*

**List of Sample Output** [show security zones on page 614](#)  
[show security zones abc on page 615](#)  
[show security zones abc detail on page 615](#)  
[show security zones terse on page 615](#)

**Output Fields** [Table 85 on page 613](#) lists the output fields for the **show security zones** command. Output fields are listed in the approximate order in which they appear.

**Table 85: 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 85: 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

## 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

```

## Sample Output

### 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
```

## Sample Output

### 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
```

## Sample Output

### show security zones terse

```
user@host> show security zones terse
Zone                Type
my-internal         Security
my-external         Security
dmz                 Security
```

## show security zones type

**Supported Platforms** [SRX Series, vSRX](#)

**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 and Interfaces Overview*
  - *Supported System Services for Host Inbound Traffic*
  - *security-zone*

**List of Sample Output** [show security zones type functional on page 617](#)  
[show security zones type security on page 617](#)  
[show security zones type security terse on page 618](#)  
[show security zones type security detail on page 618](#)

**Output Fields** [Table 86 on page 616](#) lists the output fields for the **show security zones type** command. Output fields are listed in the approximate order in which they appear.

**Table 86: 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



Table 86: show security zones type Output Fields (*continued*)

Field Name	Field Description	Level of Output
<b>Interfaces bound</b>	Number of interfaces in the zone.	none
		detail
<b>Interfaces</b>	List of the interfaces in the zone.	none
		detail
<b>Zone</b>	Name of the zone.	All levels
<b>Type</b>	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:
```