

Next Gen Services Interfaces User Guide for Routing Devices

Next Gen Services Interfaces User Guide
for Routing Devices

Published
2025-09-23

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

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

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

Next Gen Services Interfaces User Guide for Routing Devices Next Gen Services Interfaces User Guide for Routing Devices

Copyright © 2025 Juniper Networks, Inc. All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

1

About This Guide | xiii

Overview

Next Gen Services Overview | 2

Next Gen Services Overview | 2

Configuration Overview | 16

Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16

Overview | 17

Interfaces | 18

Service Set | 22

Stateful Firewall | 25

Carrier Grade Network Address Translation (CGNAT) | 32

Intrusion Detection System (IDS) | 70

Migrate from the MS Card to the MX-SPC3 | 77

Next Gen Services Feature Configuration Overview | 79

How to Configure Services Interfaces for Next Gen Services | 81

How to Configure Interface-Style Service Sets for Next Gen Services | 83

How to Configure Next-Hop Style Service Sets for Next Gen Services | 84

How to Configure Service Set Limits for Next Gen Services | 85

Example: Next Gen Services Inter-Chassis Stateful High Availability for NAT and Stateful Firewall (MX-SPC3) | 88

Requirements | 88

Overview | 88

Configuration | 88

Example: Configuring AutoVPN with Pre-Shared Key | 100

Enabling and Disabling Next Gen Services | 105

Loading the Software Images on Next-Generation Routing Engines | 105

Enabling Next Gen Services on an MX Series Router | 107

Disabling Next Gen Services on an MX Series Router | 108

Determining Whether Next Gen Services is Enabled on an MX Series Router | 109

Global System Logging Overview and Configuration | 110

Understanding Next Gen Services CGNAT Global System Logging | 110

Enabling Global System Logging for Next Gen Services | 112

Configuring Local System Logging for Next Gen Services | 113

Configuring System Logging to One or More Remote Servers for Next Gen Services | 115

System Log Error Messages for Next Gen Services | 118

Configuring Syslog Events for NAT Rule Conditions with Next Gen Services | 127

Next Gen Services SNMP MIBS and Traps | 128

Next Gen Services SNMP MIBs and Traps | 128

2

Carrier Grade NAT (CGNAT)

Deterministic NAT Overview and Configuration | 154

Deterministic NAPT Overview for Next Gen Services | 154

Configuring Deterministic NAPT for Next Gen Services | 160

Configuring the NAT Pool for Deterministic NAPT for Next Gen Services | 160

Configuring the NAT Rule for Deterministic NAPT44 for Next Gen Services | 162

Configuring the NAT Rule for Deterministic NAPT64 for Next Gen Services | 163

Configuring the Service Set for Deterministic NAT for Next Gen Services | 164

Clearing the Don't Fragment Bit | 165

Dynamic Address-Only Source NAT Overview and Configuration | 166

Dynamic Address-Only Source Translation Overview | 166

Configuring Dynamic Address-Only Source NAT for Next Gen Services | 167

Configuring the Source Pool for Dynamic Address-Only Source NAT | 167

Configuring the NAT Source Rule for Dynamic Address-Only Source NAT | 168

Configuring the Service Set for Dynamic Address-Only Source NAT | 170

Network Address Port Translation Overview and Configuration | 171

Network Address Port Translation (NAPT) Overview | 171

Configuring Network Address Port Translation for Next Gen Services | 172

- Configuring the Source Pool for NAPT | 172
- Configuring the NAT Source Rule for NAPT | 176
- Configuring the Service Set for NAPT | 178

Configuring Syslog Events for NAT Rule Conditions with Next Gen Services | 179

NAT46 | 181

NAT46 Next Gen Services Configuration Examples | 181

Stateful NAT64 Overview and Configuration | 185

Stateful NAT64 Overview | 185

IPv4 Addresses Embedded in IPv6 Addresses | 186

Configuring Next Gen Services Stateful NAT64 | 187

- Configuring the Source Pool for Stateful NAT64 | 187
- Configuring the NAT Rules for Stateful NAT64 | 191
- Configuring the Service Set for Stateful NAT64 | 194
- Clearing the Don't Fragment Bit | 194

IPv4 Connectivity Across IPv6-Only Network Using 464XLAT Overview and Configuration | 195

464XLAT Overview | 195

IPv4 Addresses Embedded in IPv6 Addresses | 197

Configuring 464XLAT Provider-Side Translator for IPv4 Connectivity Across IPv6-Only Network for Next Gen Services | 198

- Configuring the Source Pool for 464XLAT | 199
- Configuring the NAT Rules for 464XLAT | 201
- Configuring the Service Set for 464XLAT | 204
- Clearing the Don't Fragment Bit | 205

IPv6 NAT Protocol Translation (NAT PT) | 206

IPv6 NAT PT Overview | 206

IPv6 NAT-PT Communication Overview | 207

Stateless Source Network Prefix Translation for IPv6 Overview and Configuration | 209

Stateless Source Network Prefix Translation for IPv6 | 209

- Stateless Source Network Prefix Translation for IPv6 for IPv6 | 209
- Configuring NPTv6 for Next Gen Services | 210

- Configuring the Source Pool | 210

- Configuring the NAT Rule | 211

- Configuring the Service Set | 212

Transitioning to IPv6 Using Softwires | 214

6rd Softwires in Next Gen Services | 214

- 6rd Softwires in Next Gen Services Overview | 214

- Configuring Inline 6rd for Next Gen Services | 215

- Configuring a 6rd Softwire Concentrator | 215

- Configuring a 6rd Softwire Rule | 216

- Configuring Inline Services and an Inline Services Interface | 217

- Configuring the IPv4-Facing and IPv6-Facing Interfaces for 6rd | 218

- Configuring the Service Set | 219

Transitioning to IPv6 Using DS-Lite Softwires | 220

DS-Lite Softwires—IPv4 over IPv6 for Next Gen Services | 220

Configuring Next Gen Services DS-Lite Softwires | 223

- Configuring Next Gen Services Softwire Rules | 223

- Configuring Service Sets for Next Gen Services Softwires | 225

- Configuring the DS-Lite Softwire | 227

DS-Lite Subnet Limitation | 229

- DS-Lite Per Subnet Limitation Overview | 229

- Configuring DS-Lite Per Subnet Session Limitation to Prevent Denial of Service Attacks | 232

Protecting CGN Devices Against Denial of Service (DOS) Attacks | 234

Reducing Traffic and Bandwidth Requirements Using Port Control Protocol | 235

Port Control Protocol Overview | 235

Configuring Port Control Protocol | 239

- Configuring PCP Server Options | 239

- Configuring a PCP Rule | 241

- Configuring a NAT Rule | 243

- Configuring a Service Set to Apply PCP | 243

- SYSLOG Message Configuration | 244

Transitioning to IPv6 Using MAP-E and MAP-T | 245

Mapping of Address and Port with Encapsulation (MAP-E) for Next Gen Services | 245

Understanding Mapping of Address and Port with Encapsulation (MAP-E) | 245

Configuring Mapping of Address and Port with Encapsulation (MAP-E) for Next Gen Services | 249

Equal Cost Multiple Path (ECMP) support for Mapping of Address and Port with Encapsulation (MAP-E) | 252

Equal Cost Multiple Path (ECMP) support for Mapping of Address and Port with Encapsulation (MAP-E) | 253

Disabling auto-routes to support ECMP with Mapping of Address and Port with Encapsulation (MAP-E) | 253

Mapping of Address and Port with Translation (MAP-T) for Next Gen Services | 257

Understanding Mapping of Address and Port with Translation (MAP-T) | 257

Configuring Mapping of Address and Port using Translation (MAP-T) | 260

Monitoring and Troubleshooting Softwires | 265

Ping and Traceroute for DS-Lite | 265

Monitoring Softwire Statistics | 266

Monitoring CGN, Stateful Firewall, and Softwire Flows | 268

Port Forwarding Overview and Configuration | 270

Port Forwarding for Next Gen Services | 270

Port Forwarding Overview | 270

Configuring Port Forwarding with Static Destination Address Translation for Next Gen Services | 271

Configuring the Destination Pool for Destination Address Translation | 271

Configuring the Mappings for Port Forwarding | 272

Configuring the NAT Rule for Port Forwarding with Destination Address Translation | 272

Configuring the Service Set for Port Forwarding with Destination Address Translation | 274

Configuring Port Forwarding without Static Destination Address Translation for Next Gen Services | 275

Configuring the Mappings for Port Forwarding | 275

Configuring the NAT Rule for Port Forwarding without Destination Address Translation | 276

Configuring the Service Set for Port Forwarding without Destination Address Translation | 277

Port Translation Features Overview and Configuration | 279

Address Pooling and Endpoint Independent Mapping for Port Translation | 279

Round-Robin Port Allocation | 281

Secured Port Block Allocation for Port Translation | 282

Static Source NAT Overview and Configuration | 283

Static Source NAT Overview | 283

Configuring Static Source NAT44 or NAT66 for Next Gen Services | 284

Configuring the Source Pool for Static Source NAT44 or NAT66 | 284

Configuring the NAT Rule for Static Source NAT44 or NAT66 | 285

Configuring the Service Set for Static Source NAT44 or NAT66 | 286

Static Destination NAT Overview and Configuration | 288

Static Destination NAT Overview | 288

Configuring Static Destination NAT for Next Gen Services | 289

Configuring the Destination Pool for Static Destination NAT | 289

Configuring the NAT Rule for Static Destination NAT | 289

Configuring the Service Set for Static Destination NAT | 291

Twice NAPT Overview and Configuration | 293

Twice NAPT Overview | 293

Configuring Twice NAPT for Next Gen Services | 294

Configuring the Source and Destination Pools for Twice NAPT | 294

Configuring the NAT Rules for Twice NAPT | 298

Configuring the Service Set for Twice NAPT | 301

Twice NAT Overview and Configuration | 303

Twice Static NAT Overview | 303

Configuring Twice Static NAT44 for Next Gen Services | 304

Configuring the Source and Destination Pools for Twice Static NAT44 | 304

Configuring the NAT Rules for Twice Static NAT44 | 305

Configuring the Service Set for Twice Static NAT44 | 308

Twice Dynamic NAT Overview | 309

Configuring Twice Dynamic NAT for Next Gen Services | 309

Configuring the Source and Destination Pools for Twice Dynamic NAT | 310

Configuring the NAT Rules for Twice Dynamic NAT | 311

Configuring the Service Set for Twice Dynamic NAT | 314

Class of Service Overview and Configuration | 315

Class of Service for Services PICs (Next Gen Services) | 315

Class of Service Overview for Services PICs (Next Gen Services) | 315

Configuring CoS for Traffic Processed by a Services PIC (Next Gen Services) | 316

Configuring CoS Rules | 316

Configuring Application Profiles for CoS Rules | 319

Configuring CoS Rule Sets | 321

Configuring the Service Set for CoS | 321

3

Stateful Firewall Services

Stateful Firewall Services Overview and Configuration | 324

Stateful Firewall Overview for Next Gen Services | 324

Configuring Stateful Firewalls for Next Gen Services | 327

Configuring Stateful Firewall Rules for Next Gen Services | 327

Configuring Stateful Firewall Rule Sets for Next Gen Services | 330

Configuring the Service Set for Stateful Firewalls for Next Gen Services | 330

4

Intrusion Detection Services

IDS Screens for Network Attack Protection Overview and Configuration | 333

Understanding IDS Screens for Network Attack Protection | 333

Configuring Network Attack Protection With IDS Screens for Next Gen Services | 337

Configuring the IDS Screen Name, Direction, and Alarm Option | 337

Configuring Session Limits in the IDS Screen | 338

Configuring Suspicious Packet Pattern Detection in the IDS Screen | 343

Configuring the Service Set for IDS | 346

Configuring the TCP SYN cookie | 347

Overview | 348

Requirements | 348

Configuration | 348

5

Traffic Load Balancing

Traffic Load Balancing Overview and Configuration | 352

Traffic Load Balancer Overview | 352

Configuring TLB | 364

6

- Loading the TLB Service Package | 365
- Configuring a TLB Instance Name | 365
- Configuring Interface and Routing Information | 366
- Configuring Servers | 369
- Configuring Network Monitoring Profiles | 369
- Configuring Server Groups | 371
- Configuring Virtual Services | 373
- Configuring Tracing for the Health Check Monitoring Function | 376

DNS Request Filtering

DNS Request Filtering Overview and Configuration | 381

DNS Request Filtering for Disallowed Website Domains | 381

- Overview of DNS Request Filtering | 381
- How to Configure DNS Request Filtering | 384
 - How to Configure a Domain Filter Database | 384
 - How to Configure a DNS Filter Profile | 385
 - How to Configure a Service Set for DNS Filtering | 391
- Multitenant Support for DNS Filtering | 391
- Configuring Multi-tenant Support for DNS Filtering | 392
- Example: Configuring Multitenant Support for DNS Filtering | 397
 - Configuration | 398

DNS Request Filtering System Logging Error Messages | 403

7

URL Filtering

URL Filtering | 417

URL Filtering Overview | 417

Configuring URL Filtering | 423

8

Integration of Juniper ATP Cloud and Web filtering on MX Routers

Integration of Juniper ATP Cloud and Web filtering on MX Routers | 430

Integration of Juniper ATP Cloud and Web Filtering on MX Series Routers | 430

- Overview | 430
- Configuring the Web Filter Profile for Sampling | 436
- GeoIP Filtering | 441
- Global Allowlist and Global Blocklist | 443

9

Aggregated Multiservices Interfaces

Enabling Load Balancing and High Availability Using Multiservices Interfaces | 446

Understanding Aggregated Multiservices Interfaces for Next Gen Services | 446

Configuring Aggregated Multiservices Interfaces | 452

Configuring Load Balancing on AMS Infrastructure | 455

Configuring Warm Standby for Services Interfaces | 459

10

Inter-Chassis Services PIC High Availability

Inter-Chassis Services PIC High Availability Overview and Configuration | 462

Next Gen Services Inter-chassis High Availability Overview for NAT, Stateful Firewall, and IDS Flows | 462

Inter-chassis High Availability Overview for NAT, Stateful Firewall, and IDS Flows for Next Gen Services | 463

Example: Next Gen Services Inter-Chassis Stateful High Availability for NAT and Stateful Firewall (MX-SPC3) | 463

Requirements | 464

Overview | 464

Configuration | 464

Inter-Chassis Stateful Synchronization for Long Lived NAT, Stateful Firewall, and IDS Flows for Next Gen Services | 476

Inter-Chassis Stateful Synchronization Overview | 477

Configuring Inter-Chassis Stateful Synchronization for Long- Lived NAT, Stateful Firewall, and IDS Flows for Next Gen Services | 478

Configuring Inter-Chassis Stateful Synchronization for Next Gen Services with non-AMS Interface | 479

Configuring Inter-Chassis Stateful Synchronization for Next Gen Services with AMS Interface | 481

Inter-Chassis Services Redundancy Overview for Next Gen Services | 485

Configuring Inter-Chassis Services Redundancy for Next Gen Services | 488

Configuring Non-Stop Services Redundancy for Next Gen Services Service Set | 488

Configuring One-Way Services Redundancy for Next Gen Services Service Set | 495

11

Application Layer Gateways

Enabling Traffic to Pass Securely Using Application Layer Gateways | 509

Next Gen Services Application Layer Gateways | 509

Configuring Application Sets | 519

Configuring Application Properties for Next Gen Services | 520

Examples: Configuring Application Protocols | 537

Verifying the Output of ALG Sessions | 538

12

NAT, Stateful Firewall, and IDS Flows

Inline NAT Services Overview and Configuration | 551

Inline Static Source NAT Overview | 551

Configuring Inline Static Source NAT44 for Next Gen Services | 552

Configuring the Source Pool for Inline Static Source NAT44 | 552

Configuring the NAT Rule for Inline Static Source NAT44 | 553

Configuring the Service Set for Inline Static Source NAT44 | 554

Configuring Inline Services and an Inline Services Interface | 555

Inline Static Destination NAT Overview | 556

Configuring Inline Static Destination NAT for Next Gen Services | 556

Configuring the Destination Pool for Inline Static Destination NAT | 557

Configuring the NAT Rule for Inline Static Destination NAT | 557

Configuring the Service Set for Inline Static Destination NAT | 559

Configuring Inline Services and an Inline Services Interface | 559

Inline Twice Static NAT Overview | 560

Configuring Inline Twice Static NAT44 for Next Gen Services | 561

Configuring the Source and Destination Pools for Inline Twice Static NAT44 | 561

Configuring the NAT Rules for Inline Twice Static NAT44 | 562

Configuring the Service Set for Inline Twice Static NAT44 | 564

Configuring Inline Services and an Inline Services Interface | 565

13

Configuration Statements

Junos CLI Reference Overview | 568

About This Guide

Use this guide to understand and configure Next Gen Services on MX240, MX480, and MX960 routers.

1

PART

Overview

- [Next Gen Services Overview | 2](#)
 - [Configuration Overview | 16](#)
 - [Global System Logging Overview and Configuration | 110](#)
 - [Next Gen Services SNMP MIBS and Traps | 128](#)
-

CHAPTER 1

Next Gen Services Overview

IN THIS CHAPTER

- [Next Gen Services Overview | 2](#)

Next Gen Services Overview

IN THIS SECTION

- [MX Series 5G Universal Router Services Overview | 2](#)
- [Adaptive Services Overview | 3](#)
- [Next Gen Services | 4](#)
- [Summary of Services Supported on MX Series 5G Universal Routers | 4](#)
- [Next Gen Services Documentation | 7](#)
- [Enabling Next Gen Services | 8](#)
- [Compatibility with Other Services Cards | 8](#)
- [Configuring the MX-SPC3 Services Card | 10](#)
- [Methods for Applying Services to Traffic | 11](#)
- [Configuring IPsec VPN on MX-SPC3 Services Card | 11](#)

This topic provides an overview of Next Gen Services and includes the following topics

MX Series 5G Universal Router Services Overview

MX Series 5G Universal routers support several types of Services interfaces, which provide specific capabilities for inspecting, monitoring and manipulating traffic as it transits an MX Series router. Services can be categorized into Adaptive Services and Next Gen Services, with each category providing Inline

services interfaces and Multiservices interfaces options. [Table 1 on page 3](#) lists the cards that provide these services.



NOTE: The MX-SPC3 replaces MS- type cards providing a significant overall performance improvement together with high-end scale and capacity.

Table 1: MX Series 5G Universal Router Services

MX Series 5G Universal Routing Platform					
Adaptive Services				Next Gen Services	
MPC	MS-DPC	MS-MPC	MS-MIC	MPC	MX-SPC3
si-1/0/0	sp-1/0/0	ms-1/0/0	ms-1/0/0	si-1/0/0	vms-1/0/0
Inline services				Inline services	

- Adaptive Services can run on MS-DPC, MS-MPC, and MS-MIC cards using Multiservices (MS) PICs or Adaptive Services (AS) PICs.
- Next Gen Services can run on MPC cards and the MX-SPC3 security services card.

Inline services are configured on MX Series Modular Port Concentrators (MPC)s. Inline services interfaces, are virtual physical interfaces that reside on the Packet Forwarding Engine. They provide high performance processing on traffic transiting the MPC, and allow you to maximize your chassis slot capacity and utilization.

Multiservices Security cards (MS-DPC, MS-MPC, MS-MIC or MX-SPC3), provide services that can be applied to any traffic transiting the MX chassis beyond just an individual MPC. They also provide dedicated processing to support a variety of security features at scale and high performance.

Adaptive Services Overview

Adaptive Services run inline on MPCs and on MS-DPC, MS-MPC, and MS-MIC Multiservice security cards. Adaptive Services (AS) PICs and Multiservices PICs enable you to perform multiple services on the same PIC by configuring a set of services and applications. The AS and Multiservices PICs offer a range of services that you can configure in one or more service sets.



NOTE: On Juniper Networks MX Series 5G Universal Routing Platforms, the MS-DPC provides essentially the same capabilities as the MS-MPC. The interfaces on both platforms are configured in the same way.

For more information about Adaptive Services including inline services, see [Adaptive Services Overview](#).

Inline Services

Adaptive Services also use *inline services interfaces* to provide *inline* services. Inline services interfaces are virtual interfaces that reside on the Packet Forwarding Engine.

You configure inline services only on MPCs using the naming convention **si-*fpc/pic/port*** rather than the **ms-*fpc/pic/port*** naming convention.

Next Gen Services

Next Gen Services provide the combined capabilities of MX and SRX security services enabling you to inspect, monitor and manipulate traffic as it transits the MX Series router. Next Gen Services are supported both inline on Modular Port Concentrators (MPCs) and the MX-SPC3 security services card in MX240, MX480 and MX960 routers. Please refer to [Table 2 on page 5](#), which provides a summary of Next Gen Services that are supported both inline and on the MX-SPC3 card. Both Inline and MX-SPC3 based services can be used at the same time.

You configure Next Gen Services on the MX-SPC3 security services card using the *virtual multiservices* naming convention: **vms-*fpc/pic/port***.

Summary of Services Supported on MX Series 5G Universal Routers

[Table 2 on page 5](#) provides a summary of the services supported under Next Gen Services.

Table 2: Summary of Services Supported on MX Series 5G Universal Routing Platform

Next Gen Services: Inline (si-) Interface and MX-SPC3				
Service Feature	Inline Services		MX-SPC3	
	Junos OS Release	Sub-Service	Junos OS Release	Sub-Service
CGNAT	19.3R2	Basic-NAT44 and NAT66 Static Destination NAT Twice-NAT44 Basic 6rd Softwires NPTv6	19.3R2	Basic-NAT44 Basic-NAT66 Dynamic-NAT44 Static Destination NAT Basic-NAT-PT NAPT-PT NAPT44 NAPT66 Port Block Allocation Deterministic-nat44 and nat64 End Point Independent Mapping (EIM)/End Point Independent Filtering (EIF) Persistent NAT – Application Pool Pairing (APP) Twice-NAT44 – Basic, Dynamic and NAPT NAT64 XLAT-464 NPTv6

Table 2: Summary of Services Supported on MX Series 5G Universal Routing Platform (Continued)

Next Gen Services: Inline (si-) Interface and MX-SPC3				
Service Feature	Inline Services		MX-SPC3	
	Junos OS Release	Sub-Service	Junos OS Release	Sub-Service
			20.1R1	Port Control Protocol (PCP) – v1 and v2
	20.2R1	MAP-E		DS-Lite NAT46
Traffic Load Balancer	19.3R2		19.3R2	
SecIntel (ATP Cloud IP Threat Feeds)	19.3R2		N/A	
Stateful Firewall Services	N/A		19.3R2	
Intrusion Detection Services (IDS)	N/A		19.3R2	
DNS Request Filtering	N/A		19.3R2	
Aggregated Multiservices Interfaces	N/A		19.3R2	
Inter-chassis High Availability	N/A		19.3R2	CGNAT, Stateful Firewall, IDS
URL Filtering	N/A		20.1R1	

Table 2: Summary of Services Supported on MX Series 5G Universal Routing Platform (Continued)

Next Gen Services: Inline (si-) Interface and MX-SPC3				
Service Feature	Inline Services		MX-SPC3	
	Junos OS Release	Sub-Service	Junos OS Release	Sub-Service
JFlow	20.1R1		N/A	
RPM and TWAMP	20.1R1		N/A	
Video Monitoring	20.1R1		N/A	
IPsec VPN	N/A		21.1R1	Route based Site 2 Site VPN Traffic selector based VPNs AutoVPN Routing protocols (BGP/OSPF) over IPsec
Inline IPsec	24.2R1		N/A	

See [Configuring IPsec VPN on MX-SPC3 Services Card](#) for more information on IPsec support on SPC3 line card.

Next Gen Services Documentation

You can run Next Gen Services on the MX240, MX480, and MX960 if you have the MX-SPC3 services card installed in the router. Refer to our [TechLibrary](#) for all MX router documentation. For Next Gen Services, refer to the following documentation:

- To learn about and configure Next Gen Services, see *Next Gen Services Interfaces User Guide for Routing Devices* (this guide).
- For details on installing or replacing the MX-SPC3 card, see [MX Series 5G Universal Routing Platform Interface Module Reference](#).

- To monitor flows and sample traffic – See the [Monitoring, Sampling, and Collection Services Interfaces Feature Guide](#), which describes how to configure traffic flow monitoring, packet flow capture, traffic sampling for accounting or discard, port mirroring to an external device, and real-time performance monitoring.
- [Broadband Subscriber Services User Guide](#)

Enabling Next Gen Services

To run Next Gen Services, you must enable it on the MX Series router. This enables the operating system to run its own operating system (OS) for Next Gen Services.

There are specific steps you'll need to take if you're migrating your services from legacy services cards to the MX-SPC3. The Next Gen Services CLI differs from these legacy services. For more information, see ["Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3" on page 16](#).

Compatibility with Other Services Cards

The MX-SPC3 services card is compatible end-to-end with the MX Series Switch Fabrics, Routing Engines and MS-MPC line cards as described in [Table 3 on page 8](#).

Table 3: MX-SPC3 Services Card Compatibility with MX Series Switch Fabrics, Routing Engines and MPC Line Cards

Switch Fabric	Route Engine	MPC Line Cards
SCBE	RE-S-1800X4-16G-BB	MPC2E-3D
	RE-S-1800X4-16G-UPG-BB	MPC2-3D-NG
	RE-S-1800X4-16G-S	MPC3E and MPC3E-3D-NG
	RE-S-1800X4-16G-R	MPC4E-3D
	RE-S-1800X4-32G-BB	MPC-3D-16XGE
	RE-S-1800X4-32G-UB	
	RE-S-1800X4-32G-S	
	RE-S-1800X4-32G-R	

Table 3: MX-SPC3 Services Card Compatibility with MX Series Switch Fabrics, Routing Engines and MPC Line Cards *(Continued)*

Switch Fabric	Route Engine	MPC Line Cards
SCBE2	RE-S-1800X4-16G-BB	MPC2E-3D
	RE-S-1800X4-16G-UPG-BB	MPC2-3D-NG
	RE-S-1800X4-16G-S	MPC3E and MPC3E-3D-NG
	RE-S-1800X4-16G-R	MPC4E-3D
	RE-S-1800X4-32G-BB	MPC5E and MPC5EQ
	RE-S-1800X4-32G-UB	MPC7E and MPC7EQ
	RE-S-1800X4-32G-S	MPC-3D-16XGE
	RE-S-1800X4-32G-R	
	RE-S-X6-64G-BB	
	RE-S-X6-64G-UB	
	RE-S-X6-64G-S	
	RE-S-X6-64G-R	
	RE-S-X6-128G-S-BB	
	RE-S-X6-128G-S-S	
	RE-S-X6-128G-S-R	

Table 3: MX-SPC3 Services Card Compatibility with MX Series Switch Fabrics, Routing Engines and MPC Line Cards (Continued)

Switch Fabric	Route Engine	MPC Line Cards
SCBE3	RE-S-1800X4-16G-BB	MPC2-3D-NG
	RE-S-1800X4-16G-UPG-BB	MPC3E-3D-NG
	RE-S-1800X4-16G-S	MPC4E-3D
	RE-S-1800X4-16G-R	MPC5E and MPC5EQ
	RE-S-1800X4-32G-BB	MPC7E and MPC7EQ
	RE-S-1800X4-32G-UB	MPC-3D-16XGE
	RE-S-1800X4-32G-S	MPC10E-10C
	RE-S-1800X4-32G-R	MPC10E-15C
	RE-S-X6-64G-BB	
	RE-S-X6-64G-UB	
	RE-S-X6-64G-S	
	RE-S-X6-64G-R	
	RE-S-X6-128G-S-BB	
	RE-S-X6-128G-S-S	
	RE-S-X6-128G-S-R	

Configuring the MX-SPC3 Services Card

The interfaces on the MX-SPC3 services card are referred to as a virtual multi service (vms) PIC. When you configure an MX-SPC3 interface, you specify the interface as a vms- interface as follows:

```
user@host# set services service-set service-set-name interface-service service-interface vms-slot-  
number/pic-number/0.logical-unit-number
```

Aside from the CLI differences, you need to be aware of the basic hardware differences between multiservices (MS) type (MS-DPC, MS-MPC, and MS-MIC) cards and the MX-SPC3 services card. MS type cards contain four CPU complexes whereas the MX-SPC3 card, while more powerful, contains two CPU complexes. Each CPU complex services a single PIC, meaning that MS type cards support four PICs

whereas the MX-SPC3 supports two PICs. MS type cards use special multiservices (MS) and adaptive services (AS) PICs, whereas the PICs on the MX-SPC3 card are integrated.

Because the number of PICs directly affects the number of interfaces, you might need to add logical units to each interface on the MX-SPC3 to increase the number of interfaces to four. For example, if you currently use all four interfaces on the MS type card and you have a service set per interface, you can create two logical units per interface on the MX-SPC3 to bring the total number of interfaces to four, and then reassociate the four service sets to these four logical interfaces.

Methods for Applying Services to Traffic

When you configure Next Gen Services, you can apply those services with either of the following methods:

- Apply the configured services to traffic that flows through a particular interface on the MX router.
- Apply the configured services to traffic that is destined for a particular next hop.

Configuring IPsec VPN on MX-SPC3 Services Card

To configuring IPsec on MX-SPC3 service card, use the CLI configuration statements at the [edit security] hierarchy level as the IPsec CLI configuration at the [edit services] is replaced with the CLI configuration at the [edit security] hierarchy level as shown in [Table 4 on page 11](#)

Table 4: Comparison on configuring IPsec VPN for MX and MX-SPC3

Current MX Configuration	Equivalent MX-SPC3 Configuration
set services ipsec-vpn traceoptions	set security ike traceoptions
set services ipsec-vpn ike proposal	set security ike proposal
set services ipsec-vpn ike policy	set security ike policy
set services ipsec-vpn ike policy <i>policy-name</i> respond-bad-spi	set security ike respond-bad-spi
set services ipsec-vpn ipsec proposal	set security ipsec proposal
set services ipsec-vpn ipsec policy	set security ipsec policy

Table 4: Comparison on configuring IPsec VPN for MX and MX-SPC3 (Continued)

Current MX Configuration	Equivalent MX-SPC3 Configuration
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> from [source-address] destination-address]	set security ipsec vpn <i>vpn-name</i> traffic-selector <i>selector-name</i> [local-ip remote-ip]
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> from ipsec-inside-interface	set security ipsec vpn <i>vpn-name</i> bind-interface
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then remote-gateway	set security ike gateway <i>gw-name</i> address
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then backup-remote-gateway	set security ike gateway <i>gw-name</i> address
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then dead-peer-detection	set security ike gateway <i>gw-name</i> dead-peer-detection
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then dynamic ike-policy	set security ike gateway <i>gw-name</i> ike-policy
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then dynamic ipsec-policy	set security ipsec vpn <i>vpn-name</i> ike ipsec-policy
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then manual	set security ipsec vpn <i>vpn-name</i> manual
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then clear-dont-fragment-bit	set security ipsec vpn <i>vpn-name</i> df-bit clear
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then copy-dont-fragment-bit	set security ipsec vpn <i>vpn-name</i> df-bit copy
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then set-dont-fragment-bit	set security ipsec vpn <i>vpn-name</i> df-bit copy
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then tunnel-mtu	set security ipsec vpn <i>vpn-name</i> tunnel-mtu
set services ipsec-vpn rule <i>rule-name</i> term <i>term-name</i> then no-anti-replay	set security ipsec vpn <i>vpn-name</i> ike no-anti-replay

Table 4: Comparison on configuring IPsec VPN for MX and MX-SPC3 (Continued)

Current MX Configuration	Equivalent MX-SPC3 Configuration
set services ipsec-vpn rule <i>rule-name</i> match-direction	set security ipsec vpn <i>vpn-name</i> match-direction
set services ipsec-vpn establish-tunnels	set security ipsec vpn <i>vpn-name</i> establish-tunnels
set services service-set <i>svc-set-name</i> ipsec-vpn-options local-gateway <i>address</i>	set security ipsec vpn <i>vpn-name</i> ike gateway <i>gateway-name</i>
set services service-set <i>svc-set-name</i> ipsec-vpn-options clear-dont-fragment-bit	No global service-set setting. Must be configured on a per vpn object basis.
set services service-set <i>svc-set-name</i> ipsec-vpn-options copy-dont-fragment-bit	No global service-set setting. Must be configured on a per vpn object basis.
set services service-set <i>svc-set-name</i> ipsec-vpn-options set-dont-fragment-bit	No global service-set setting. Must be configured on a per vpn object basis.
set services service-set <i>svc-set-name</i> ipsec-vpn-options udp-encapsulate	set security ipsec vpn <i>vpn-name</i> udp-encapsulate
set services service-set <i>svc-set-name</i> ipsec-vpn-options no-anti-replay	No global service-set setting. Must be configured on a per vpn object basis.
set services service-set <i>svc-set-name</i> ipsec-vpn-options passive-mode-tunneling	set security ipsec vpn <i>vpn-name</i> passive-mode-tunneling
set services service-set <i>svc-set-name</i> ipsec-vpn-options tunnel-mtu	No global service-set setting. Must be configured on a per vpn object basis.
set services service-set <i>svc-set-name</i> ipsec-vpn-rules	set services service-set <i>svc-set-name</i> ipsec-vpn-rules
set services ipsec-vpn rule <rule-name> term <term-name> then tunnel-mtu	set security ipsec vpn <vpn-name> tunnel-mtu

Understanding Tunnel MTU

The MTU for st0 is at the interface level. With tunnel-MTU feature we achieve tunnel level MTU. With Tunnel-MTU feature we can configure MTU at the VPN object level. You can configure tunnel-mtu to

control tunnel MTU, if st0 MTU or IFL MTU is not configured it will impact the MTU behaviour. The minimum Tunnel MTU you can configure for IPv6 traffic is 1390.

Tunnel MTU feature is not supported on PMI (Power mode IPsec). Tunnel-mtu configuration is at VPN hierarch and not at the traffic selector level, hence the tunnel-mtu configuration applies to all the tunnels (all TS) belonging to that VPN. Tunnel MTU config change is considered as catastrophic change (deletes existing tunnel). Configuration change of no-icmp-packet-too-big is not considered as catastrophic.

Pre-fragmentation is done considering IPsec tunnel overhead of minimum tunnel MTU configuration or AMS outside IFL MTU. Post-fragmentation requires MTU to be set on the external interface and the corresponding IPsec counters do not increment for egress traffic. Post fragmentation is done by IOC and not by MX-SPC3 card. In MX-SPC3, the default st0 MTU for inet and inet6 family is 9192, there is no default value for tunnel-mtu configuration at VPN hierarchy. IPv6 packets are fragmented at source host and not fragmented at intermediate routers so pre-fragmentation does not apply for IPv6 packets.

For IPv4 packets, the pre-fragmentation, post-fragmentation, and ICMP Fragmentation needed and DF set error occurs in following cases:

- When the inner packet length is lesser than the difference of tunnel-mtu and tunnel overhead then no fragmentation occurs.
- When the inner packet length is greater than the difference of tunnel-mtu and tunnel overhead, and the inner packet DF bit is not set then pre-fragmentation occurs.
- When the inner packet length is greater than the difference of tunnel-mtu and tunnel overhead, and the outer tunnel DF bit is not set then encapsulation, and post-fragmentation occurs.
- When the inner packet length is greater than the difference of tunnel-mtu and tunnel overhead, and both the inner packet DF bit and outer tunnel DF bit is set then packet is dropped and ICMP Fragmentation Needed and DF Set error sent back.

For IPv6 packets, the pre-fragmentation, post-fragmentation, and ICMP Packet Too Big error occurs in following cases:

- When the inner packet length is lesser than the difference of tunnel-mtu and tunnel overhead then no fragmentation occurs.
- When the inner packet length is greater than the difference of tunnel-mtu and tunnel overhead, and the outer tunnel DF bit is not set then encapsulation, and post-fragmentation occurs.
- When the inner packet length is greater than the difference of tunnel-mtu and tunnel overhead, and the outer tunnel DF bit is set then packet is dropped and if no-icmp-packet-too-big is not set then ICMP Packet Too Big error sent.

- When the inner packet length is greater than the difference of tunnel-mtu and tunnel overhead, and the outer tunnel DF bit is set then packet is dropped and if no-icmp-packet-too-big is set then ICMP Packet Too Big error is not sent

Difference between st0 MTU and tunnel MTU

- Tunnel-MTU is at different level compared to st0 MTU.
- st0 MTU is interface level MTU and tunnel-MTU feature achieves tunnel level MTU
- In MX-SPC3, PFE checks st0 mtu to fragment or drop the packet. Hence, packet does not reach flowd or IPsec and will not have any control over the MTU action.
- VPN tunnel-mtu configuration value is less than the st0 MTU.

RELATED DOCUMENTATION

[Enabling and Disabling Next Gen Services | 105](#)

[Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16](#)

[Adaptive Services Overview](#)

CHAPTER 2

Configuration Overview

IN THIS CHAPTER

- [Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16](#)
- [Next Gen Services Feature Configuration Overview | 79](#)
- [How to Configure Services Interfaces for Next Gen Services | 81](#)
- [How to Configure Interface-Style Service Sets for Next Gen Services | 83](#)
- [How to Configure Next-Hop Style Service Sets for Next Gen Services | 84](#)
- [How to Configure Service Set Limits for Next Gen Services | 85](#)
- [Example: Next Gen Services Inter-Chassis Stateful High Availability for NAT and Stateful Firewall \(MX-SPC3\) | 88](#)
- [Example: Configuring AutoVPN with Pre-Shared Key | 100](#)
- [Enabling and Disabling Next Gen Services | 105](#)

Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3

IN THIS SECTION

- [Overview | 17](#)
- [Interfaces | 18](#)
- [Service Set | 22](#)
- [Stateful Firewall | 25](#)
- [Carrier Grade Network Address Translation \(CGNAT\) | 32](#)
- [Intrusion Detection System \(IDS\) | 70](#)
- [Migrate from the MS Card to the MX-SPC3 | 77](#)

Overview

Next Gen Services on the MX-SPC3 require you to configure services differently from what you are accustomed to with Adaptive Services, which run on MS type cards (MS-MPC, MS-MIC and MS-DPC). Configuring the MX-SPC3 services card more closely aligns with the way you configure the SRX Series services gateway. Once you are familiar with this more unified approach, you should be able to configure services on these two platforms in a more seamless fashion, ultimately resulting in less training overhead and lower risk of configuration error.

Aside from the CLI differences, you need to be aware of the basic hardware differences between multiservices (MS) type (MS-DPC, MS-MPC, and MS-MIC) cards and the MX-SPC3 services card. MS type cards contain four CPU complexes whereas the MX-SPC3 card, while more powerful, contains two CPU complexes. Each CPU complex services a single PIC, meaning that MS type cards support four PICs whereas the MX-SPC3 supports two PICs. MS type cards use special multiservices (MS) and adaptive services (AS) PICs, whereas the PICs on the MX-SPC3 card are integrated.

Because the number of PICs directly affects the number of interfaces ([Table 5 on page 17](#)), you might need to add logical units to each interface on the MX-SPC3 to increase the number of interfaces to four. For example, if you currently use all four interfaces on the MS type card and you have a service set per interface, you can create two logical units per interface on the MX-SPC3 to bring the total number of interfaces to four, and then reassociate the four service sets to these four logical interfaces.

Table 5: Hardware Differences: MS Type Cards versus MX-SPC3 Card

	MS-Cards	MX-SPC3
Number of CPU complexes	4	2
Number of PICs per CPU complex	1	1
Number of interfaces per PIC	1	1
Total number of interfaces on card	4	2

NOTE: See the [MX Series 5G Universal Routing Platform Interface Module Reference](#) for more information on the MX-SPC3 hardware.

The following sections provide an overview of the basic configuration differences between services on the MS type cards and services on the MX-SPC3 card. The intent of these sections is to help you get started by using basic examples to illustrate the major changes. These examples show a subset of the CLI configuration options and do not replace the more formal treatment of the subject matter found in

the Next Gen Services Interfaces User Guide for Routing Devices and the Junos OS CLI Reference Guide.

The configuration examples in these sections are presented side-by-side so you can easily see the differences between the two. The examples are intended to show you how to configure existing MS type card features on the MX-SPC3. The examples are not intended to show you how to configure new features only found on the MX-SPC3. For legibility and ease of comparison, the order of statements presented might differ slightly from the actual order of statements displayed in the CLI.

If you have a large set of existing adaptive services, we recognize that these changes might be an inconvenience to you. To help you migrate from MS type cards to the MX-SPC3, we suggest that you proceed as follows:

- Look through the examples in this guide to get an overall view of the changes required.
- Look through the set of configuration examples in knowledge base article KB35348.
- Look through this guide and the Junos OS CLI Reference Guide to understand all the features, configuration options, and syntax.
- Contact JTAC for help with your migration.

You do not need to make these configuration changes if you continue to run adaptive services on the MS type cards. However, once you deploy the MX-SPC3 on a router, you must replace all MS type cards on that router and reconfigure your services to align with the Next Gen Services configuration paradigm.

Interfaces

MS type cards use the interface naming convention `ms-1/0/0`, whereas you specify MX-SPC3 interfaces using the virtual multiservices or `vms-1/0/0` interface naming convention. There are no changes to the names of `ams` and `mams` interfaces.

In addition, a number of parameters that are configured under `services-options` on an `ms` interface are configured under `service-set-options` in a service set.

[Table 6 on page 19](#) shows examples of these changes.

Table 6: Interfaces and Service Options

MS Type Cards	MX-SPC3
<pre>[edit interfaces] ms-5/1/0 { <...> }</pre>	<pre>[edit interfaces] # Change interface name to vms. vms-5/1/0 { <...> }</pre>
<pre>[edit interfaces] ms-5/1/0 { services-options { open-timeout 40; close-timeout 40; inactivity-tcp-timeout 10; inactivity-asymm-tcp-timeout 10; tcp-tickles 8; ignore-errors tcp; } }</pre>	<pre>[edit services] service-set sset1 { service-set-options { # Set tcp parameters under tcp-session. tcp-session { open-timeout 40; close-timeout 40; inactivity-tcp-timeout 10; inactivity-asymm-tcp-timeout 10; tcp-tickles 8; ignore-errors tcp; } } }</pre>
<pre>[edit interfaces] ms-5/1/0 { services-options { inactivity-non-tcp-timeout 40; session-timeout 10; } }</pre>	<pre>[edit services] service-set sset1 { # Set non-tcp parameters directly under # service-set-options. service-set-options { inactivity-non-tcp-timeout 40; session-timeout 10; } }</pre>

Table 6: Interfaces and Service Options (*Continued*)

MS Type Cards	MX-SPC3
<pre>[edit interfaces] ms-5/1/0 { services-options { fragment-limit 10; reassembly-timeout 5; } }</pre>	<pre>[edit interfaces] vms-5/1/0 { services-options { fragment-limit 10; reassembly-timeout 5; } }</pre>
<pre>[edit interfaces] ms-5/1/0 { services-options { session-limit { maximum 100; cpu-load-threshold 12; rate 10; } } }</pre>	<pre>[edit services] # Maximum number of sessions can be # specified per service-set. service-set sset1 { service-set-options { session-limit { maximum 100; } } }</pre> <p>[edit interfaces]</p> <p># All session-limit parameters continue to be # configurable per interface. If the maximum # number of sessions is different from the associated # service-set, the smaller number takes effect.</p> <pre>vms-5/1/0 { services-options { session-limit { maximum 100; cpu-load-threshold 12; rate 10; } } }</pre>

Table 6: Interfaces and Service Options *(Continued)*

MS Type Cards	MX-SPC3
<pre>[edit interfaces] ms-5/1/0 { services-options { pba-interim-logging-interval 10; } }</pre>	<pre>[edit interfaces] # Set interim-logging-interval under the nat branch. nat { source { pool src-pool { port { block-allocation { interim-logging-interval 10; } } } } }</pre>
<pre>[edit interfaces] ms-5/1/0 { services-options { syslog { host { <...> } } } }</pre>	<p>See service-set syslog stream host.</p>
<pre>[edit interfaces] ms-5/1/0 { services-options { syslog { message-rate-limit 10; } } }</pre>	<pre>[edit services] service-set sset1 { syslog { event-rate 10; } }</pre>

Table 6: Interfaces and Service Options (Continued)

MS Type Cards	MX-SPC3
<pre> [edit interfaces] ms-5/1/0 { services-options { ignore-errors alg; disable-global-timeout-override; trio-flow-offload { minimum-bytes 1000; } } } </pre>	Not supported

Service Set

[Table 7 on page 22](#) shows minor changes in the way some service-set parameters are configured.

Table 7: Service Set

MS Type Cards	MX-SPC3
<pre> [edit services] service-set sset1 { tcp-mss 1460; service-set-options { tcp-non-syn drop-flow-send-rst; tcp-fast-open drop; } } </pre>	<pre> [edit services] service-set sset1 { service-set-options { # Set tcp parameters under tcp-session. tcp-session { tcp-mss 1460; tcp-non-syn drop-flow-send-rst; tcp-fast-open drop; } } } </pre>

Table 7: Service Set (*Continued*)

MS Type Cards	MX-SPC3
<pre>[edit services] service-set sset1 { replicate-services { replication-threshold 180; } }</pre>	<pre>[edit interfaces] # Set replication-threshold on the interface. vms-5/1/0 { redundancy-options { replication-threshold 180; } }</pre>
<pre>[edit services] service-set sset1 { syslog { host 10.1.1.1 { port 514; } } }</pre>	<pre>[edit services] service-set sset1 { syslog # Process security logs in the dataplane. mode stream; stream s1 { # Specify host to send security logs to. host { 10.1.1.1; port 514; } } } }</pre>

Table 7: Service Set (*Continued*)

MS Type Cards	MX-SPC3
<pre>[edit services] service-set sset1 { syslog { host local; } }</pre>	<pre>[edit services] service-set sset1 { syslog # Process security logs in the control plane, # saving logs to local file specified by rtlog. mode event; } } rtlog { traceoptions { # Specify filename for logs. file rtlog size 1g; flag all; } }</pre>
<pre>[edit services] service-set sset1 { service-order <...> }</pre>	<p>Service order is fixed.</p>
<pre>[edit services] service-set sset1 { sampling-service <...> }</pre>	<p>J-Flow logging is supported inline.</p>

Table 7: Service Set (Continued)

MS Type Cards	MX-SPC3
<pre>[edit services] service-set sset1 { tag-rule-sets <...> tag-rules <...> hcm-profile <...> hcm-url-rule-sets <...> hcm-url-rules <...> service-set-options { bypass-traffic-on-pic-failure; } }</pre>	Currently unsupported

Stateful Firewall

IN THIS SECTION

- [Rules and Policies | 25](#)
- [Address Lists and Ranges | 28](#)
- [Applications | 31](#)
- [Traceoptions and Counters | 31](#)

Rules and Policies

Stateful firewall rules on the MX-SPC3 are structured slightly differently from stateful firewall rules for services on the MS type cards. On the MX-SPC3, you enclose the rules within a policies wrapper, and you define the match terms and actions for the rule in a policy contained within the rule.

Just like a stateful firewall service on the MS type card, you create a service set to associate an interface with a rule set. A rule set contains references to one or more rules. Rules are applied sequentially in the order that you list them until a match occurs and an action taken.

Each rule contains one or more pairs of match terms and actions. On the MX-SPC3, each pair of match terms and actions is called a policy. Policies are applied sequentially in the order that you specify them until a match occurs and an action taken.

Table 8 on page 26 shows the configuration differences between stateful firewall rules on the MS card and the MX-SPC3. In particular, note the different definitions for the permit/deny/reject actions.

Table 8: Stateful Firewall Rules and Policies

MS Card	MX-SPC3
[edit services]	[edit services]
<pre> service-set s1 { stateful-firewall-rule-sets rule-set- basic-sfw; interface-service { service-interface ms-1/1/0; } } </pre>	<pre> service-set s1 { policies stateful-firewall-rule-sets rule-set-basic-sfw; interface-service { service-interface vms-1/1/0; } } </pre>
<pre> stateful-firewall { </pre>	<pre> # Enclose stateful firewall rules within the policies wrapper. policies { </pre>

Table 8: Stateful Firewall Rules and Policies *(Continued)*

MS Card	MX-SPC3
<pre> rule Rule1 { match-direction input; term ping-https-apps { from { source-address { any } destination-address { any } applications [junos-icmp- ping junos-https]; } then { accept/reject/discard skip-ids; syslog; } } term accept { then { accept; } } } # end Rule1 </pre>	<pre> policies stateful-firewall-rule Rule1 { match-direction input; # Define match terms and actions in a policy. policy ping-https-apps { # Unlike the from statement, the match statement (and # source-address, destination-address, and application) # are mandatory. match { source-address any; destination-address any; application [junos-icmp-ping junos-https]; } then { # permit = allow # deny = silently drop # reject = drop and send ICMP unreachable or TCP RST permit/deny/reject # skip-ids is not supported. One possible way of # achieving this same goal is to create two # service-sets, one with IDS and one without IDS, # and route your next-hop-service # traffic to the desired service set via the associated # inside or outside interface. log; } } policy accept { match { source-address any; destination-address any; application any; } then { permit; </pre>

Table 8: Stateful Firewall Rules and Policies *(Continued)*

MS Card	MX-SPC3
	<pre> } } } # end Rule1 </pre>
<pre> rule Rule2 { match-direction output; term local { from { source-address { 10.1.3.2/32; } application-sets APPL-SET1; } then { accept; } } } # end Rule2 </pre>	<pre> policies stateful-firewall-rule Rule2 { match-direction output; policy local { match { source-address 10.1.3.2/32; destination-address any; # application can refer to an application set. application APPL-SET1; } then { permit; } } } # end Rule2 </pre>
<pre> rule-set rule-set-basic-sfw { rule Rule1; rule Rule2; } } # end stateful-firewall </pre>	<pre> # Use the stateful-firewall-rule-set element to list the # firewall rules in the order that you want them applied. stateful-firewall-rule-set rule-set-basic-sfw { stateful-firewall-rule Rule1; stateful-firewall-rule Rule2; } } # end policies </pre>

Address Lists and Ranges

Stateful firewall rules can contain match terms that refer to address ranges and lists.

On the MS card, you use `source-address-range` and `destination-address-range` elements to specify address ranges and `prefix-list` elements under `policy-options` to specify address lists. The `prefix-list` element is

not for use solely for stateful firewall rules. You also use the `prefix-list` element to specify address lists for use within routing policies.

On the MX-SPC3, the `prefix-list` element is not used for stateful firewall rules. You use an `address-book` under `services` to define address lists and ranges for use within stateful firewall rules. The `prefix-list` element still exists, but is used exclusively for routing policies. You therefore need to configure both `address-book` and `prefix-list` elements if you are specifying address lists for stateful firewall rules and address lists for routing policies.

[Table 9 on page 30](#) shows the differences between how you specify addresses for stateful firewall rules on the MS card versus the MX-SPC3.

Table 9: Addresses

MS Card	MX-SPC3
<pre> [edit] policy-options { prefix-list p1 { 10.1.22.45/32; 192.168.0.11/32; } } [edit services] stateful-firewall { rule sfw-rule { match-direction input; term banned-addresses { from { source-prefix-list { p1; } source-address-range { low 10.1.22.100 high 10.1.22.109; } } } then { reject; syslog; } } } <...> </pre>	<pre> [edit services] # Define address lists and address ranges in an address book. address-book { global { address-set p1 { address p1-a; address p1-b; } address p1-a 10.1.22.45/32; address p1-b 192.168.0.11/32; address p2 { address-range 10.1.22.100/32 { to { 10.1.22.109/32; } } } } } # end address-book policies { stateful-firewall-rule sfw-rule { match-direction input; policy banned-addresses { match { # Refer to the addresses defined in the address book. source-address [p1 p2]; destination-address any; application any; } then { deny; log; } } } } <...> </pre>

Applications

The MX-SPC3 supports more built-in Junos applications than the MS card. You can match on these built-in applications when you create a stateful firewall rule.

To see the complete list of built-in applications, use the `show groups junos-defaults applications` configuration mode command. For example:

```
[edit]
# show groups junos-defaults applications | match junos
application junos-ftp {
  application junos-ftp-data {
    application junos-tftp {
      application junos-twamp {
        application junos-rtsp {
          application junos-netbios-session {

<...>
```

Traceoptions and Counters

Stateful firewalls for Next Gen Services on the MX-SPC3 support additional capabilities to help debug and count traffic:

- `traceoptions` - Use to trace policy-related events such as policy lookups and rules-based events. The events are captured in the specified file for viewing.
- `count` - Use to count traffic-related events such as incoming/outgoing bytes and packets. View the counters using show commands:
 - `show services policies detail` - the output includes traffic-related counters when you specify the `count` option in your policy
 - `show services policies hit-count` - the hit count is always available regardless of whether you use the `count` option in your policy or not

[Table 10 on page 32](#) shows how to use the `traceoptions` and `count` elements:

Table 10: Traceoptions and Count

MS Card	MX-SPC3
Not supported	<pre> [edit services] policies { # Enable traceoptions to trace policy-related events. traceoptions { file policylogs size 10m files 5; flag all; } stateful-firewall-rule Rule1 { match-direction input; policy my-policy { match { source-address any; destination-address any; application [junos-dns-udp junos-dns-tcp]; } then { permit # Enable counting of traffic events. count; } } } # end my-policy ... </pre>

Carrier Grade Network Address Translation (CGNAT)

Configuring NAT for Next Gen Services on the MX-SPC3 is different from configuring NAT on legacy services on the MS card in a number of ways:

- On the MX-SPC3, you configure source NAT separately from destination NAT. You configure source NAT in the source branch of the configuration tree and you configure destination NAT in the destination branch of the configuration tree. Source NAT and destination NAT each has its own sets of address pools and rules in its respective branch of the configuration tree.
- On the MX-SPC3, if you configure both source NAT and destination NAT, destination NAT applies first, and then source NAT applies to the destination NAT translated result. In other words, you write the source NAT rule not based on the original packet, but based on the destination NAT translated result.

- On the MX-SPC3, you do not explicitly configure a translation-type. The type of translation is determined implicitly by your configuration.
- On the MX-SPC3, port translation is the default behavior for dynamic mappings (where different pre-NAT addresses might map to the same post-NAT address over time). If you do not explicitly include the port statement in a pool definition, port translation takes place with a port range [1024, 65535], and the port is selected in a round robin fashion. If you do not want port translation to take place, you must add a port statement with the no-translation option. This default does not apply to static mappings where a pre-NAT address always maps to the same post-NAT address.

Table 11 on page 33 through Table 23 on page 64 show examples of how the different translation types are configured on the MX-SPC3.

Table 11: Example: Basic-NAT44

MS Card	MX-SPC3
[edit services]	[edit services]
<pre>service-set sset1 { nat-rules rule-basic-nat44; interface-service { service-interface ms-1/2/0; } }</pre>	<pre>service-set sset1 { nat-rule-sets rule-basic-nat44; interface-service { service-interface vms-2/0/0; } }</pre>
<pre>nat {</pre>	<pre>nat { source {</pre>

Table 11: Example: Basic-NAT44 (Continued)

MS Card	MX-SPC3
<pre>pool src-pool { address 10.10.10.0/24; }</pre>	<pre>pool src-pool { address { 10.10.10.0/24; } # host-address-base indicates a type of static mapping # where the base address 10.45.1.0/0 maps to the # lowest address in the pool, namely 10.10.10.0/0, # and the other addresses map sequentially from there # e.g. 10.45.1.1 maps to 10.10.10.1, and so on. # Since this is a static mapping, there is no port translation # by default. # Note that host-address-base does not have to be the # lowest address allowed by the subsequent source rule. # Any packet with a source address allowed by the source rule # but is lower than the host-address-base is discarded. host-address-base 10.45.1.0/0; }</pre>

Table 11: Example: Basic-NAT44 (Continued)

MS Card	MX-SPC3
<pre>rule rule-basic-nat44 { match-direction input; term t1 { from { source-address { 10.45.1.0/24 } } then { translated { source-pool src-pool; translation-type { basic-nat44; } } } } }</pre>	<pre>rule-set rule-basic-nat44 { match-direction input; rule r1 { match { source-address 10.45.1.0/24; } then { source-nat { pool { src-pool; } } } } }</pre>
<pre>} # end nat</pre>	<pre> } # end source } # end nat</pre>

Table 12: Example: Basic-NAT66

MS Card	MX-SPC3
<pre>[edit services]</pre>	<pre>[edit services]</pre>

Table 12: Example: Basic-NAT66 (Continued)

MS Card	MX-SPC3
<pre> service-set sset1 { nat-rules rule-basic-nat66; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-basic-nat66; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>
<pre> pool src-pool { address 2001:DB8:2222::0/96; } </pre>	<pre> pool src-pool { address { 2001:DB8:2222::0/96; } } </pre>

Table 12: Example: Basic-NAT66 (Continued)

MS Card	MX-SPC3
<pre> rule rule-basic-nat66 { match-direction input; term t1 { from { source-address { 2001:DB8:1111::0/96; } } then { translated { source-pool src-pool; translation-type { basic-nat66; } } } } } </pre>	<pre> rule-set rule-basic-nat66 { match-direction input; rule r1 { match { source-address 2001:DB8:1111::0/96; } then { source-nat { pool { src-pool; } } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source } # end nat </pre>

Table 13: Example: Dynamic-NAT44

MS Card	MX-SPC3
<pre> [edit services] </pre>	<pre> [edit services] </pre>

Table 13: Example: Dynamic-NAT44 (Continued)

MS Card	MX-SPC3
<pre> service-set sset1 { nat-rules rule-dynamic-nat44; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-dynamic-nat44; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>
<pre> pool src-pool { address-range low 10.10.10.2 high 10.10.10.10; } </pre>	<pre> pool src-pool { address { 10.10.10.2/32 to 10.10.10.10/32; } # Since this is implicitly a dynamic mapping, # there is port translation by default , so we need to # explicitly specify that we don't want port translation. port { no-translation; } } </pre>

Table 13: Example: Dynamic-NAT44 (Continued)

MS Card	MX-SPC3
<pre> rule rule-dynamic-nat44 { match-direction input; term t0 { from { applications junos-icmp-all; } then { no-translation; } } term t1 { from { destination-address { 10.99.0.2/32; } source-address-range { low 10.45.0.2 high 10.45.0.10; } } then { translated { source-pool src-pool; translation-type { dynamic-nat44; } } } } } </pre>	<pre> rule-set rule-dynamic-nat44 { match-direction input; rule r0 { match { source-address 0.0.0.0/0; application junos-icmp-all; } then { source-nat { off; } } } rule r1 { match { source-address-name addr1; destination-address 10.99.0.2/32; } then { source-nat { pool { src-pool; } } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source } # end nat </pre>

Table 13: Example: Dynamic-NAT44 (*Continued*)

MS Card	MX-SPC3
	<pre> address-book { global { address addr1 { address-range 10.45.0.2/32 { to { 10.45.0.10/32; } } } } } </pre>

Table 14: Example: NAPT-44

MS Card	MX-SPC3
[edit services]	[edit services]
<pre> service-set sset1 { nat-rules rule-napt44; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-napt44; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>

Table 14: Example: NAPT-44 (Continued)

MS Card	MX-SPC3
<pre> pool src-pool { address 10.10.10.0/24; port { automatic; } } </pre>	<pre> pool src-pool { address { 10.10.10.0/24; } # Since this is implicitly a dynamic mapping, # and there is no explicit port statement # to indicate otherwise, the default port # mapping behavior takes effect. } </pre>
<pre> rule rule-napt44 { match-direction input; term t1 { from { source-address { 10.45.1.0/24 } application-sets accept-algs; } then { translated { source-pool src-pool; translation-type { napt44; } } } } } </pre>	<pre> rule-set rule-napt44 { match-direction input; rule r1 { match { source-address 10.45.1.0/24; application accept-algs; } then { source-nat { pool { src-pool; } } } } } </pre>

Table 14: Example: NAPT-44 (Continued)

MS Card	MX-SPC3
<pre> } # end nat </pre>	<pre> } # end source } # end nat </pre>

Table 15: Example: napt-66

MS Card	MX-SPC3
<pre> [edit services] </pre>	<pre> [edit services] </pre>
<pre> service-set sset1 { nat-rules rule-napt66; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-napt66; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>

Table 15: Example: napt-66 (Continued)

MS Card	MX-SPC3
<pre> pool src-pool { address 2001:DB8:2222::0/112; port { range low 20000 high 30000; } } </pre>	<pre> pool src-pool { address { 2001:DB8:2222::0/112; } port { range { 20000; to { 30000; } } } } </pre>
<pre> rule rule-napt66 { match-direction input; term t1 { from { source-address { 2001:DB8:1111::0/96; } } then { translated { source-pool src-pool; translation-type { napt66; } } } } } </pre>	<pre> rule-set rule-napt66 { match-direction input; rule r1 { match { source-address 2001:DB8:1111::0/96; } then { source-nat { pool { src-pool; } } } } } </pre>

Table 15: Example: napt-66 (Continued)

MS Card	MX-SPC3
<pre> } # end nat </pre>	<pre> } # end source } # end nat </pre>

Table 16: Example: Deterministic NAT-44

MS Card	MX-SPC3
<pre> [edit services] </pre>	<pre> [edit services] </pre>
<pre> service-set sset1 { nat-rules rule-dnat-44; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-dnat-44; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { destination { </pre>
<pre> pool dest-pool { address 10.10.10.2/32; } </pre>	<pre> pool dest-pool { address { 10.10.10.2/32; } } </pre>

Table 16: Example: Deterministic NAT-44 (Continued)

MS Card	MX-SPC3
<pre> rule rule-dnat-44 { match-direction input; term t1 { from { destination-address { 10.45.0.2/32 } } then { translated { destination-pool dest-pool; translation-type { dnat-44; } } } } } </pre>	<pre> rule-set rule-dnat-44 { match-direction input; rule r1 { match { destination-address 10.45.0.2/32; } then { destination-nat { pool { dest-pool; } } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end destination } # end nat </pre>

Table 17: Example: Stateful-NAT464

MS Card	MX-SPC3
[edit services]	[edit services]

Table 17: Example: Stateful-NAT464 (Continued)

MS Card	MX-SPC3
<pre> service-set sset1 { nat-rules rule-stateful-nat464; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-stateful-nat464-src; nat-rule-sets rule-stateful-nat464-dest; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>
<pre> pool src-pool { address 10.10.10.0/24; port { automatic; } } } </pre>	<pre> pool src-pool { address { 10.10.10.0/24; } port { automatic { round-robin; } } } } </pre>

Table 17: Example: Stateful-NAT464 (Continued)

MS Card	MX-SPC3
<pre> rule rule-stateful-nat464 { match-direction input; term t1 { from { source-address { 2001:DB8:1111::0/96; } destination-address { 2001:DB8:2222::0/96; } applications [junos-icmp- all junos-icmp-ping junos-traceroute junos- traceroute-ttl 1]; } then { translated { source-pool src-pool; clat-prefix 2001:DB8:1111::0/96; destination-prefix 2001:DB8:2222::0/96; translation-type { stateful-nat464; } } } } } </pre>	<pre> # This source rule applies after the destination rule. rule-set rule-stateful-nat464-src { match-direction input; rule r1 { match { source-address 2001:DB8:1111::0/96; # Since destination NAT happens first, the # destination IPv6 prefix has been stripped off, # resulting in an IPv4 destination address. destination-address 0.0.0.0/0; application [junos-icmp-all junos-icmp-ping junos-traceroute junos-traceroute-ttl 1]; } then { source-nat { pool { src-pool; } clat-prefix 2001:DB8:1111::0/96; } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source </pre>

Table 17: Example: Stateful-NAT464 (Continued)

MS Card	MX-SPC3
	<pre> destination { </pre>
	<pre> # This destination rule applies before the source rule. rule-set rule-stateful-nat464-dest { match-direction input; rule r1 { match { destination-address 2001:DB8:2222::0/96; } then { destination-nat { destination-prefix 2001:DB8:2222::0/96; } } } } </pre>
	<pre> } # end destination } # end nat </pre>

Table 18: Example: Stateful-NAT64

MS Card	MX-SPC3
[edit services]	[edit services]

Table 18: Example: Stateful-NAT64 (Continued)

MS Card	MX-SPC3
<pre> service-set sset1 { nat-rules rule-stateful-nat64; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-stateful-nat64-src; nat-rule-sets rule-stateful-nat64-dest; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>
<pre> pool src-pool { address 10.10.10.0/24; port { automatic; random-allocation; } } mapping-timeout 500; } </pre>	<pre> pool src-pool { address { 10.10.10.0/24; } port { automatic { random-allocation; } } mapping-timeout 500; } } </pre>

Table 18: Example: Stateful-NAT64 (Continued)

MS Card	MX-SPC3
<pre> rule rule-stateful-nat64 { match-direction input; term t1 { from { destination-address { 2001:DB8:2222::0/64; } } then { translated { source-pool src-pool; destination-prefix 2001:DB8:2222::0/64; translation-type { stateful-nat64; } } } } term t2 { from { destination-address { 2001:DB8:3333::0/64; } } then { translated { source-pool src-pool; destination-prefix 2001:DB8:3333::0/64; translation-type { stateful-nat64; } } } } } </pre>	<pre> # This source rule applies after the destination rule. rule-set rule-stateful-nat64-src { match-direction input; rule r1 { match { source-address 0::/0; # Since destination NAT applies first, the # destination address is now IPv4. destination-address 0.0.0.0/0; } then { source-nat { pool { src-pool; } } } } } </pre>

Table 18: Example: Stateful-NAT64 (Continued)

MS Card	MX-SPC3
<pre>} # end nat</pre>	<pre>} # end source</pre>
	<pre>destination {</pre>
	<pre># This destination rule applies before the source rule. rule-set rule-stateful-nat64-dest { match-direction input; rule r1 { match { destination-address 2001:DB8:2222::0/64; } then { destination-nat { destination-prefix 2001:DB8:2222::0/64; } } } rule r2 { match { destination-address 2001:DB8:3333::0/64; } then { destination-nat { destination-prefix 2001:DB8:3333::0/64; } } } }</pre>

Table 18: Example: Stateful-NAT64 (Continued)

MS Card	MX-SPC3
	<pre> } # end destination } # end nat </pre>

Table 19: Example: Twice-Basic-NAT-44

MS Card	MX-SPC3
[edit services]	[edit services]
<pre> service-set sset1 { nat-rules rule-twice-basic-nat-44; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-twice-basic-nat-44-src; nat-rule-sets rule-twice-basic-nat-44-dest; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>

Table 19: Example: Twice-Basic-NAT-44 (Continued)

MS Card	MX-SPC3
<pre>pool src-pool { address 10.98.10.0/24; } pool dest-pool { address 10.99.10.0/24; }</pre>	<pre>pool src-pool { address { 10.98.10.0/24; } # host-address-base indicates a type of static mapping where # the base address 10.10.10.0/0 maps to the lowest # address in the pool, namely 10.98.10.0/0, # and the other addresses map sequentially from there # e.g. 10.10.10.1 maps to 10.98.10.1, and so on. # Since this is a static mapping, there is no port translation # by default. # Note that host-address-base does not have to be the # lowest address allowed by the subsequent source rule. # Any packet with a source address allowed by the source rule # but is lower than the host-address-base is discarded. host-address-base 10.10.10.0/0; } }</pre>

Table 19: Example: Twice-Basic-NAT-44 (Continued)

MS Card	MX-SPC3
<pre> rule rule-twice-basic-nat-44 { match-direction input; term t1 { from { source-address { 10.10.10.0/24; } destination-address { 10.20.10.0/24; } } then { translated { source-pool src- pool; destination-pool dest-pool; translation-type { twice-basic- nat-44; } } } } } </pre>	<pre> # This source rule applies after the destination rule. rule-set rule-twice-basic-nat-44-src { match-direction input; rule r1 { match { source-address 10.10.10.0/24; # Since destination NAT happens first, the # address refers to the NAT'd address. destination-address 10.99.10.0/24; } then { source-nat { pool { src-pool; } } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source </pre>
	<pre> destination { </pre>

Table 19: Example: Twice-Basic-NAT-44 (Continued)

MS Card	MX-SPC3
	<pre> pool dest-pool { address { 10.99.10.0/24; } } </pre>
	<pre> # This destination rule applies before the source rule. rule-set rule-twice-basic-nat-44-dest { match-direction input; rule r1 { match { destination-address 10.20.10.0/24; } then { destination-nat { pool { dest-pool; } } } } } </pre>
	<pre> } # end destination } # end nat </pre>

Table 20: Example: Twice-Dynamic-NAT-44

MS Card	MX-SPC3
[edit services]	[edit services]
<pre> service-set sset1 { nat-rules rule-twice-dynamic-nat-44; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-twice-dynamic-nat-44-src; nat-rule-sets rule-twice-dynamic-nat-44-dest; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>
<pre> pool src-pool { address 10.98.10.0/24; } pool dest-pool { address 10.99.10.0/24; } </pre>	<pre> pool src-pool { address { 10.98.10.0/24; } port { no-translation; } } </pre>

Table 20: Example: Twice-Dynamic-NAT-44 (Continued)

MS Card	MX-SPC3
<pre> rule rule-twice-dynamic-nat-44 { match-direction input; term t1 { from { source-address { 10.10.10.0/24; } destination-address { 10.20.10.0/24; } } then { translated { source-pool src-pool; destination-pool dest- pool; translation-type { twice-dynamic- nat-44; } } } } } </pre>	<pre> # This source rule applies after the destination rule. rule-set rule-twice-dynamic-nat-44-src { match-direction input; rule r1 { match { source-address 10.10.10.0/24; # Since destination NAT happens first, the destination # address refers to the NAT'd address. destination-address 10.99.10.0/24; } then { source-nat { pool { src-pool; } } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source </pre>
	<pre> destination { </pre>

Table 20: Example: Twice-Dynamic-NAT-44 (Continued)

MS Card	MX-SPC3
	<pre> pool dest-pool { # By default, address mapping in destination pools is static. address { 10.99.10.0/24; } } </pre>
	<pre> # This destination rule applies before the source rule. rule-set rule-twice-dynamic-nat-44-dest { match-direction input; rule r1 { match { destination-address 10.20.10.0/24; } then { destination-nat { pool { dest-pool; } } } } } </pre>
	<pre> } # end destination } # end nat </pre>

Table 21: Example: Twice-NAPT-44

MS Card	MX-SPC3
[edit services]	[edit services]
<pre> service-set sset1 { nat-rules rule-twice-napt-44; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-twice-napt-44-src; nat-rule-sets rule-twice-napt-44-dest; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>
<pre> pool src-pool { address 10.98.10.0/24; port { automatic; secured-port-block-allocation } block-size 256 max-blocks-per-address 1 active-block-timeout 300; } pool dest-pool { address 10.99.10.2/32; } </pre>	<pre> pool src-pool { address { 10.98.10.0/24; } port { automatic { round-robin; } block-allocation { block-size 256; maximum-blocks-per-host 1; active-block-timeout 300; } } } </pre>

Table 21: Example: Twice-NAPT-44 (Continued)

MS Card	MX-SPC3
<pre> rule rule-twice-napt-44 { match-direction input; term t1 { from { source-address { 10.10.10.0/24; } destination-address { 10.20.10.2/32; } } then { translated { source-pool src-pool; destination-pool dest- pool; translation-type { twice-napt-44; } } } } } </pre>	<pre> # This source rule applies after the destination rule. rule-set rule-twice-napt-44-src { match-direction input; rule r1 { match { source-address 10.10.10.0/24; # Since destination NAT happens first, the # destination address refers to the NAT'd address. destination-address 10.99.10.2/32; } then { source-nat { pool { src-pool; } } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source </pre>
	<pre> destination { </pre>

Table 21: Example: Twice-NAPT-44 (Continued)

MS Card	MX-SPC3
	<pre> pool dest-pool { address { 10.99.10.2/32; } } </pre>
	<pre> # This destination rule applies before the source rule. rule-set rule-twice-napt-44-dest { match-direction input; rule r1 { match { source-address 10.10.10.0/24; destination-address 10.20.10.2/32; } then { destination-nat { pool { dest-pool; } } } } } </pre>
	<pre> } # end destination } # end nat </pre>

Table 22: Example: Deterministic-NAPT44

MS Card	MX-SPC3
[edit services]	[edit services]
<pre>service-set sset1 { nat-rules rule-deterministic-napt44; interface-service { service-interface ms-1/2/0; } }</pre>	<pre>service-set sset1 { nat-rule-sets rule-deterministic-napt44; interface-service { service-interface vms-2/0/0; } }</pre>
<pre>nat {</pre>	<pre>nat { source {</pre>

Table 22: Example: Deterministic-NAPT44 (Continued)

MS Card	MX-SPC3
<pre>pool src-pool { address 10.10.10.0/24; port { range low 1024 high 19999; deterministic-port-block-allocation block-size 256; } mapping-timeout 120; }</pre>	<pre>pool src-pool { address { 10.10.10.0/24; } port { range { 1024; to { 19999; } } deterministic { block-size 256; # host address specifies the subnet that you # want to apply to this pool. host address 10.2.0.0/20; } } mapping-timeout 120; }</pre>

Table 22: Example: Deterministic-NAPT44 (Continued)

MS Card	MX-SPC3
<pre> rule rule-deterministic-napt44 { match-direction input; term t1 { from { source-address { 10.2.0.0/18; } } then { translated { source-pool src-pool; translation-type { deterministic-napt44; } mapping-type endpoint- independent; } } } } </pre>	<pre> rule-set rule-deterministic-napt44 { match-direction input; rule r1 { match { source-address 10.2.0.0/18; } then { source-nat { pool { src-pool; } mapping-type endpoint-independent; } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source } # end nat </pre>

Table 23: Example: Deterministic-NAPT64

MS Card	MX-SPC3
[edit services]	[edit services]

Table 23: Example: Deterministic-NAPT64 (Continued)

MS Card	MX-SPC3
<pre> service-set sset1 { nat-rules rule-deterministic-napt64; interface-service { service-interface ms-1/2/0; } } </pre>	<pre> service-set sset1 { nat-rule-sets rule-deterministic-napt64-src; nat-rule-sets rule-deterministic-napt64-dest; interface-service { service-interface vms-2/0/0; } } </pre>
<pre> nat { </pre>	<pre> nat { source { </pre>
<pre> pool src-pool { address 10.98.10.0/24; port { automatic; random-allocation; } deterministic-port-block- allocation block-size 256; } } } </pre>	<pre> pool src-pool { address { 10.98.10.0/24; } port { automatic { random-allocation; } deterministic { block-size 256; host address 2001:DB8:1111::1/120; } } } } } </pre>

Table 23: Example: Deterministic-NAPT64 (Continued)

MS Card	MX-SPC3
<pre> rule rule-deterministic-napt64 { match-direction input; term t1 { from { source-address { 2001:DB8:1111::1/120; } } then { translated { destination-prefix 2001:DB8:2222::/96; source-pool src-pool; translation-type { deterministic- napt64; } } } } } </pre>	<pre> # This source rule applies after the destination rule. rule-set rule-deterministic-napt64-src { match-direction input; rule r1 { match { source-address 2001:DB8:1111::1/120; # Since destination NAT happens first, the destination # address refers to the NAT'd address. destination-address 0.0.0.0/0; } then { source-nat { pool { src-pool; } } } } } </pre>
<pre> } # end nat </pre>	<pre> } # end source </pre>
	<pre> destination { </pre>

Table 23: Example: Deterministic-NAPT64 (Continued)

MS Card	MX-SPC3
	<pre> pool dest-pool { address { 10.99.10.2/32; } } </pre>
	<pre> # This destination rule applies before the source rule. rule-set rule-destination-napt64-dest { match-direction input; rule r1 { match { destination-address 2001:DB8:2222::/96; } then { destination-nat { destination-prefix 2001:DB8:2222::/96; } } } } </pre>
	<pre> } # end destination } # end nat </pre>

Table 24: Example: napt-pt

MS Card	MX-SPC3
[edit services]	[edit services]

Table 24: Example: napt-pt (Continued)

MS Card	MX-SPC3
<pre>service-set sset1 { nat-rules rule-napt-pt; interface-service { service-interface ms-1/2/0; } }</pre>	<pre>service-set sset1 { nat-rule-sets rule-napt-pt-src; nat-rule-sets rule-napt-pt-dest; interface-service { service-interface vms-2/0/0; } }</pre>
<pre>nat {</pre>	<pre>nat { source {</pre>
<pre> pool src-pool { address 10.10.10.2/32; } pool dest-pool { address 10.99.10.2/32; } }</pre>	<pre> pool src-pool { address { 10.10.10.2/32; } } }</pre>

Table 24: Example: napt-pt (Continued)

MS Card	MX-SPC3
<pre>rule rule-napt-pt { match-direction input; term t1 { from { source-address { 2001:DB8:1111::2/128; } destination-address { 2001:DB8:2222::2/128; } } then { translated { source-pool src-pool; destination-pool dest- pool; translation-type { napt-pt; } } } } }</pre>	<pre>rule-set rule-napt-pt-src { match-direction input; rule r1 { match { source-address 2001:DB8:1111::2/128; destination-address 10.99.10.0/24; } then { source-nat { pool { src-pool; } } } } }</pre>
<pre>} # end nat</pre>	<pre>} # end source</pre>
	<pre>destination {</pre>

Table 24: Example: napt-pt (Continued)

MS Card	MX-SPC3
	<pre>pool dest-pool { address { 10.99.10.2/32; } }</pre>
	<pre>rule-set rule-napt-pt-dest { match-direction input; rule r1 { match { destination-address 2001:DB8:2222::2/128; } then { destination-nat { pool { dest-pool; } } } } }</pre>
	<pre> } # end destination } # end nat</pre>

Intrusion Detection System (IDS)

IDS rules for Next Gen Services on the MX-SPC3 are defined under the screen branch. There are minor differences in the naming of the various elements, but the main change is in the behavior for detecting packets with IPv4 options and IPv6 extensions:

- For the IDS service on the MS Card, the default behavior is to detect and drop packets with IPv4 options and IPv6 extensions. If you want to allow these packets, you have to allow them explicitly through configuration.
- For the IDS Next Gen Service on the MX-SPC3, the default behavior is to allow packets with IPv4 options and IPv6 extensions. If you want to detect and drop these packets, you have to disallow them explicitly through configuration.

Table 25 on page 71 shows examples of the configuration differences.

Table 25: IDS Rules

MS Card	MX-SPC3
<pre>[edit services] service-set sset1 { ids-rules r1; ids-rules r2; }</pre>	<pre>[edit services] service-set sset1 { # Replace ids-rules with ids-option. ids-option ids1; ids-option ids2; }</pre>
<pre>[edit services] ids { rule r1 { match-direction input; term t1 { <...> } } }</pre>	<pre>[edit services] # Define ids rules under the screen branch. screen { # Replace rule with ids-option. ids-option ids1 { match-direction input; # Flatten hierarchy by removing term and placing # contents directly under ids-option. <...> } }</pre>

Table 25: IDS Rules (Continued)

MS Card	MX-SPC3
<pre>[edit services] ids { rule r1 { match-direction input; term t1 { then { allow-ip-options [loose-source-route route-record router-alert security stream-id strict- source-route timestamp]; } } } }</pre>	<pre>[edit services] screen { ids-option ids1 { match-direction input; # By default, all ip options are allowed. } }</pre>
<pre>[edit services] ids { rule r1 { match-direction input; term t1 { then { <no allow-ip-options configured> } } } }</pre>	<pre>[edit services] screen { ids-option ids1 { match-direction input; # Explicitly specify the disallowed options. ip { loose-source-route-option; record-route-option; security-option; stream-option; strict-source-route-option; timestamp-option; # router-alert option for IPv4 is not supported. } } }</pre>

Table 25: IDS Rules *(Continued)*

MS Card	MX-SPC3
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { allow-ipv6-extension-header [ah dstopts esp fragment hop-by-hop mobility routing]; } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; # By default, all ipv6 extensions are allowed. } } </pre>
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { <no allow-ipv6-extension-header configured> } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; ip { # Explicitly specify the disallowed extensions. ipv6-extension-header { AH-header; ESP-header; fragment-header; hop-by-hop-header; mobility-header; routing-header; # dstoptions is not supported. } } } } </pre>

Table 25: IDS Rules *(Continued)*

MS Card	MX-SPC3
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { aggregation { source-prefix 24; destination-prefix 24; source-prefix-ipv6 64; destination-prefix-ipv6 64; } } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; aggregation { source-prefix-mask 24; destination-prefix-mask 24; source-prefix-v6-mask 64; destination-prefix-v6-mask 64; } } } </pre>
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { icmp-fragment-check; icmp-large-packet-check; } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; # Group icmp checks under icmp. icmp { fragment; large; } } } </pre>

Table 25: IDS Rules *(Continued)*

MS Card	MX-SPC3
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { land-attack-check; tcp-winnuke-check; tcp-syn-fragment-check; tcp-syn-defense; } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; # Group tcp checks under tcp. tcp { land; winnuke; syn-frag; # tcp-syn-defense is not supported. } } } </pre>
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { session-limit { by-source { maximum 100; rate 10; packets 1k; } by-destination { maximum 100; rate 10; packets 1k; } } } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; limit-session { by-source { maximum-sessions 100; session-rate 10; packet-rate 1k; } by-destination { maximum-sessions 100; session-rate 10; packet-rate 1k; } } } } </pre>

Table 25: IDS Rules *(Continued)*

MS Card	MX-SPC3
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { session-limit { by-source { by-protocol { tcp { maximum 100; rate 10; packets 1k; } udp { maximum 100; rate 10; packets 1k; } icmp { maximum 100; rate 10; packets 1k; } } } } } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; limit-session { by-source { by-protocol { tcp { maximum-sessions 100; session-rate 10; packet-rate 1k; } udp { maximum-sessions 100; session-rate 10; packet-rate 1k; } icmp { maximum-sessions 100; session-rate 10; packet-rate 1k; } } } } } } </pre>

Table 25: IDS Rules (Continued)

MS Card	MX-SPC3
<pre> [edit services] ids { rule r1 { match-direction input; term t1 { then { session-limit { by-destination { by-protocol { tcp { maximum 100; rate 10; packets 1k; } udp { maximum 100; rate 10; packets 1k; } icmp { maximum 100; rate 10; packets 1k; } } } } } } } } </pre>	<pre> [edit services] screen { ids-option ids1 { match-direction input; limit-session { by-destination { by-protocol { tcp { maximum-sessions 100; session-rate 10; packet-rate 1k; } udp { maximum-sessions 100; session-rate 10; packet-rate 1k; } icmp { maximum-sessions 100; session-rate 10; packet-rate 1k; } } } } } } </pre>

Migrate from the MS Card to the MX-SPC3

Use this procedure to configure a router to support Next Gen Services.

You typically use this procedure to migrate a router supporting legacy services on the MS card to a router supporting Next Gen Services on the MX-SPC3, but this procedure applies even if the router that you are migrating from does not contain MS card cards.

Because Next Gen Services configuration is not compatible with legacy service provisioning, migrating a router to support Next Gen Services on the MX-SPC3 requires you to completely deprovision and reprovision your router . Furthermore:

- You cannot install an MX-SPC3 card in a router that has MS cards.
- You cannot configure Next Gen Services on a router equipped with MS cards.
- You cannot configure legacy services on a router equipped with MX-SPC3 cards.

In other words, a router can run with either MS cards or MX-SPC3 cards but not both at the same time.



NOTE: This procedure is service affecting. You are setting the router to factory default configuration.

1. Upgrade the router to release 19.3R2.
2. Back up the current router configuration to a remote host.
3. Set the router to factory default configuration.
 - a. Load the router with the factory default configuration:

```
root# load factory-default
```

- b. Configure the management interface with the same IP address as you had before you loaded the factory default configuration:

```
root# set interfaces fxp0 unit 0 family inet address <mgt-ip-address>
```

- c. Configure SSH so that you can continue to access the router. For example:

```
root# set system services ssh root-login allow
root# set system services ssh max-sessions-per-connection 32
root# set system root-authentication plain-text-password
New password:
Retype new password:
```

- d. Commit the changes.

4. Enable Next Gen Services on the router.

Junos OS provides a system-wide operational parameter that you enable if you want to configure Next Gen Services on a router. By default, this parameter is not enabled.

From operational mode:

```
root> request system enable unified-services
Before enabling unified services, please move to baseline configuration.
Are above conditions satisfied ? [yes,no]
```



NOTE: This setting is persistent and survives a reboot.

5. Reboot the router.

```
root> request system reboot
```

6. Replace the MS card cards with MX-SPC3 cards.

7. Reprovision your router.

As a starting point, you can restore the backup from step 2 but you might need to change this configuration to be compatible with Next Gen Services before you can commit.

SEE ALSO

[Next Gen Services Overview | 2](#)

[Enabling and Disabling Next Gen Services | 105](#)

Next Gen Services Feature Configuration Overview

IN THIS SECTION

- [Service Rules and Rule Sets | 80](#)
- [Service Sets | 80](#)
- [Services Interfaces | 80](#)

To configure services with Next Gen Services, you need to configure the following objects:

- Service rules

- Service sets
- Services interfaces

Service Rules and Rule Sets

Service rules specify a set of matching conditions and a set of actions to apply to traffic when it matches the conditions. For example, a stateful firewall rule can specify a destination address that must be matched, and take the action of dropping packets that have that destination address.

Service rule sets consist of a group of services rules that belong to the same category. For example, a stateful firewall rule set consists of stateful firewall rules.

Service Sets

A service set specifies one or more service rules or rule sets to apply to traffic. The service set also specifies a services interface, which indicates where the services processing is performed.

A service set is either an interface-style service set or a next-hop-style service set.

Interface-Style Service Set

The service set applies the service rules to all traffic that flows through a particular interface.

Next-Hop-Style Service Set

The service set applies the service rules to traffic that is destined for a particular next hop. You must redirect the next-hop traffic to the services interface that the service set uses.

Services Interfaces

A services interface indicates where a service is applied to traffic. Services interfaces are not physical links to external devices.

If a service is performed on an MX-SPC3 services card, the service interface has the format:

```
vms-slot-number/pic-number/port-number
```

If a service is performed on a line card's PFE (inline services), the service interface has the format *si-slot-number/pic-number/0*.

RELATED DOCUMENTATION

[Next Gen Services Overview | 2](#)

[How to Configure Services Interfaces for Next Gen Services | 81](#)

[Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16](#)

How to Configure Services Interfaces for Next Gen Services

To configure services interfaces:

1. Configure the services interface name.

```
[edit]
user@host# set interfaces interface-name
```

Where the *interface-name* one of the following:

- *vms-slot-number/pic-number/port-number* for an MX-SPC3 services card
- *si-slot-number/pic-number/0* for a line card PFE (inline services interface)

2. Configure the unit and family for the interface.

- a. If you are using the services interface in an interface service set:

```
[edit]
user@host# set interfaces interface-name unit logical-unit-number family (inet | inet6)
```

- b. If you are using the services interface in a next-hop service set, configure inside and outside interface units:

```
[edit]
user@host# set interfaces interface-name unit logical-unit-number family (inet | inet6)
user@host# set interfaces interface-name unit logical-unit-number service-domain inside
user@host# set interfaces interface-name unit logical-unit-number family (inet | inet6)
user@host# set interfaces interface-name unit logical-unit-number service-domain outside
```

For example:

```
[edit]
user@host# set interfaces vms-1/0/0 unit 100 family inet
user@host# set interfaces vms-1/0/0 unit 100 service-domain inside
user@host# set interfaces vms-1/0/0 unit 1000 family inet
user@host# set interfaces vms-1/0/0 unit 1000 service-domain outside
```

3. When neither NAT nor the `max-sessions-per-subscriber` statement at the `[edit service-set service-set-name service-set-options]` hierarchy level are configured, enable the creation of subscribers if you want to track subscribers.

```
[edit interfaces interface-name services-options]
user@host# set enable-subscriber-analysis
```

4. Configure CPU resource restrictions for the services interface.

```
[edit interfaces interface-name services-options session-limit]
user@host# set cpu-load-threshold percentage
```

When the CPU usage exceeds the value (percentage of the total available CPU resources), the system reduces the rate of new sessions so that the existing sessions are not affected by low CPU availability. The CPU utilization is constantly monitored, and if the CPU usage remains above the configured `cpu-load-threshold` value for a continuous period of 5 seconds, Junos OS reduces the session rate value configured at `edit interfaces interface-name services-options session-limit rate` by 10%. This is repeated until the CPU utilization comes down to the configured limit.

5. Configure the maximum number of sessions allowed simultaneously on a services card.

```
[edit interfaces interface-name services-options session-limit]
user@host# set maximum number
```

If you specify the maximum number of sessions to be zero, it indicates that the configuration is not effective. You must specify a value higher than zero for the maximum number of sessions.

6. Configure the maximum number of new sessions allowed per second on a services card.

```
[edit interfaces interface-name services-options session-limit]
user@host# set rate rate
```

RELATED DOCUMENTATION

[Next Gen Services Overview | 2](#)

[How to Configure Next-Hop Style Service Sets for Next Gen Services | 84](#)

[How to Configure Service Set Limits for Next Gen Services | 85](#)

[Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16](#)

How to Configure Interface-Style Service Sets for Next Gen Services

To configure an interface service set:

1. Configure the service set name.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Specify the service interface that the service set uses to apply services.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

3. Specify the service rules that the service set applies to traffic.

For example:

```
[edit services service-set ss1]
user@host# set nat-rule-sets internal-nat
```

4. (Optional) Enable the service set to process unidirectional traffic.

```
[edit services service-set service-set-name service-set-options]
user@host# set enable-asymmetric-traffic-processing
```

5. Enable service-processing at routing engine (RE).

```
[edit services service-set service-set-name service-set-options]
user@host# set routing-engine-services
```


6. Apply the service set to an interface that is passing traffic. You can apply a service filter to apply the service set to only certain traffic on the interface.

```
[edit interfaces interface-name unit logical-unit-number family (inet | inet6) service]
user@host# set (input | output) service-set service-set-name <service-filter filter-name>
```

For details about configuring the service-filter, see *Guidelines for Configuring Service Filters*.

The input option applies the service set to the input side of the interface, and the output option applies the service set to the output side of the interface. If you are using a bidirectional service rule in the service set, then the same service set must be used for input and output.

RELATED DOCUMENTATION

[Next Gen Services Overview | 2](#)

[How to Configure Interface-Style Service Sets for Next Gen Services | 83](#)

[How to Configure Service Set Limits for Next Gen Services | 85](#)

[Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16](#)

How to Configure Next-Hop Style Service Sets for Next Gen Services

To configure a next-hop service set:

1. Configure the service set name.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Specify the services interface inside unit and outside unit for the service set.

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name.unit-number outside-
service-interface interface-name.unit-number
```

The inside-service-interface must be a service interface logical unit that is configured with service-domain inside. The outside-service-interface must be a service interface logical unit that is configured with service-domain outside.

3. Specify the service rules that the service set applies to traffic.

For example:

```
[edit services service-set SS1]
user@host# set nat-rule-sets internal-nat
```

4. (Optional) Enable the service set to process unidirectional traffic.

```
[edit services service-set service-set-name service-set-options]
user@host# set enable-asymmetric-traffic-processing
```

5. Configure a static route to force traffic to the inside or outside interface of the next-hop service set.
For example, if you want traffic with the destination 198.51.100.33 to be processed by the service set with the inside interface vms-1/0/0.100:

```
[edit routing-options]
user@host# set static route 198.51.100.33 next-hop vms-1/0/0.100
```

RELATED DOCUMENTATION

[Next Gen Services Overview | 2](#)

[How to Configure Interface-Style Service Sets for Next Gen Services | 83](#)

[How to Configure Service Set Limits for Next Gen Services | 85](#)

[Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16](#)

How to Configure Service Set Limits for Next Gen Services

To configure service set limits:

1. Set the maximum number of session setups allowed per second for the service set. After this setup rate is reached, any additional session setup attempts are dropped. If you do not include the `max-session-creation-rate` statement, the session setup rate is not limited.

```
[edit services service-set service-set-name ]
user@host# set max-session-setup-rate (number | numberk)
```

If you use the *numberk* format, 1k=1000.

2. Enable packets to bypass without creating a new session when the flow in the service set exceeds the limit that is set by the `max-flows` statement at the `[edit services service-set service-set-name]` hierarchy level.

```
[edit services service-set service-set-name service-set-options]
user@host# bypass-traffic-on-exceeding-flow-limits
```

3. To limit the session open information in your system logs, you can disable it from being collected.

```
[edit services service-set service-set-name service-set-options]
user@host# set disable-session-open-syslog
```

4. Configure the maximum number of sessions allowed from a single subscriber.

```
[edit services service-set service-set-name service-set-options]
user@host# set max-sessions-per-subscriber session-number
```

5. Specify the maximum number of sessions allowed simultaneously on the service set. If you specify the maximum number of sessions to be zero, it indicates that the configuration is not effective. You must specify a value higher than zero for the maximum number of sessions.

```
[edit services service-set service-set-name service-set-options]
user@host# set session-limit maximum number
```

6. Configure the session lifetime for the service set in seconds. The session is closed after this amount of time, even if traffic is running on the session.

```
[edit services service-set service-set-name service-set-options]
user@host# set session-timeout seconds
```

7. Specify the inactivity timeout period for non-TCP established sessions.

```
user@host# set inactivity-non-tcp-timeout seconds
```

8. Configure the TCP session parameters for the service-set.

- a. Set the timeout period for the Transmission Control Protocol (TCP) session tear-down.

```
[edit services service-set-name services-options]
user@host# set close-timeout seconds
```

The default value is 1 second. The range is 2 through 300 seconds.

- b. Configure the inactivity timeout period for asymmetric TCP established sessions

```
[edit services service-set service-set-name service-set-options tcp-session]
user@host# set inactivity-asymm-tcp-timeout seconds
```

- c. Configure the number of seconds that a unidirectional TCP session can be inactive before it is closed.

```
[edit services service-set service-set-name service-set-options tcp-session]
user@host# set inactivity-tcp-timeout seconds
```

The default value is 30 seconds. The range is 4 through 86,400 seconds. Any value you configure in the application protocol definition overrides the value specified here; for more information, see ["Configuring Application Properties for Next Gen Services" on page 520](#).

- d. Set the timeout period for Transmission Control Protocol (TCP) session establishment, for use with SYN-cookie defenses against network intrusion.

```
[edit services service-set-name service-set-options ]
user@host# set open-timeout seconds
```

The default value is 5 seconds. The range of possible values is from 4 through 224 seconds. Any value you configure in the intrusion detection service (IDS) definition overrides the value specified here; for more information, see ["Configuring Network Attack Protection With IDS Screens for Next Gen Services" on page 337](#).

RELATED DOCUMENTATION

[Next Gen Services Overview | 2](#)

[How to Configure Interface-Style Service Sets for Next Gen Services | 83](#)

[Next Gen Services Feature Configuration Overview | 79](#)

Example: Next Gen Services Inter-Chassis Stateful High Availability for NAT and Stateful Firewall (MX-SPC3)

IN THIS SECTION

- [Requirements | 88](#)
- [Overview | 88](#)
- [Configuration | 88](#)

This example shows how to configure Next Gen Services inter-chassis high availability for stateful firewall and NAT services.

Requirements

This example uses the following hardware and software components:

- Two MX480 routers with MX-SPC3 services cards
- Junos OS Release 19.3R2, 19.4R1 or later

Overview

Two MX 3D routers are identically configured to facilitate stateful failover for firewall and NAT services in case of a chassis failure.

Configuration

IN THIS SECTION

- [CLI Quick Configuration | 89](#)
- [Configuring Interfaces for Chassis 1. | 91](#)
- [Configure Routing Information for Chassis 1 | 93](#)
- [Configuring NAT and Stateful Firewall for Chassis 1 | 94](#)

- [Configuring the Service Set | 96](#)
- [Configuring Interfaces for Chassis 2 | 98](#)
- [Configure Routing Information for Chassis 2 | 99](#)

To configure inter-chassis high availability for this example, perform these tasks:

CLI Quick Configuration

To quickly configure this example on the routers, copy the following commands and paste them into the router terminal window after removing line breaks and substituting interface information specific to your site.



NOTE: The following configuration is for chassis 1.

```
[edit]
set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.2
set interfaces vms-4/0/0 redundancy-options routing-instance HA
set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.1/32
set interfaces vms-4/0/0 unit 20 family inet
set interfaces vms-4/0/0 unit 20 service-domain inside
set interfaces vms-4/0/0 unit 30 family inet
set interfaces vms-4/0/0 unit 30 service-domain outside
set interfaces ge-2/0/0 vlan-tagging
set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.1/24
set routing-instances HA instance-type vrf
set routing-instances HA interface ge-2/0/0.0
set routing-instances HA interface vms-4/0/0.10
set routing-instances HA route-distinguisher 1:1
set policy-options policy-statement dummy term 1 then reject
set routing-instances HA vrf-import dummy
set routing-instances HA vrf-export dummy
set routing-instances HA routing-options static route route 5.5.5.1/32 next-hop vms-4/0/0.10
set routing-instances HA routing-options static route route 5.5.5.2/32 next-hop 20.1.1.2
set services nat pool p2 address 32.0.0.0/24
set services nat pool p2 port automatic random-allocation
set services nat pool p2 address-allocation round-robin
```

```

set services nat rule r2 match-direction input
set services nat rule r2 term t1 from source-address 129.0.0.0/8
set services nat rule r2 term t1 from source-address 128.0.0.0/8
set services nat rule r2 term t1 then translated source-pool p2
set services nat rule r2 term t1 then translated translation-type napt-44
set services nat rule r2 term t1 then translated address-pooling paired
set services nat rule r2 term t1 then syslog
set services stateful-firewall rule r2 match-direction input
set services stateful-firewall rule r2 term t1 from source-address any-unicast
set services stateful-firewall rule r2 term t1 then accept
set services stateful-firewall rule r2 term t1 then syslog
set services service-set ss2 replicate-services replication-threshold 180
set services service-set ss2 replicate-services stateful-firewall
set services service-set ss2 replicate-services nat
set services service-set ss2 stateful-firewall-rules r2
set services service-set ss2 nat-rules r2
set services service-set ss2 next-hop-service inside-service-interface vms-4/0/0.20
set services service-set ss2 next-hop-service outside-service-interface vms-4/0/0.30
set services service-set ss2 syslog host local class session-logs
set services service-set ss2 syslog host local class stateful-firewall-logs
set services service-set ss2 syslog host local class nat-logs

```



NOTE: The following configuration is for chassis 2. The NAT, stateful firewall, and service-set information must be identical for chassis 1 and 2.

```

set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.1
set interfaces vms-4/0/0 redundancy-options routing-instance HA
set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.2/32
set interfaces vms-4/0/0 unit 20 family inet
set interfaces vms-4/0/0 unit 20 service-domain inside
set interfaces vms-4/0/0 unit 30 family inet
set interfaces vms-4/0/0 unit 30 service-domain outside
set interfaces ge-2/0/0 vlan-tagging
set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.2/24
set routing-instances HA instance-type vrf
set routing-instances HA interface ge-2/0/0.0
set routing-instances HA interface vms-4/0/0.10
set routing-instances HA route-distinguisher 1:1
set policy-options policy-statement dummy term 1 then reject

```

```

set routing-instances HA vrf-import dummy
set routing-instances HA vrf-export dummy
set routing-instances HA routing-options static route 5.5.5.2/32 next-hop vms-4/0/0.10
set routing-instances HA routing-options static route 5.5.5.1/32 next-hop 20.1.1.1
set services nat pool p2 address 32.0.0.0/24
set services nat pool p2 port automatic random-allocation
set services nat pool p2 address-allocation round-robin
set services nat rule r2 match-direction input
set services nat rule r2 term t1 from source-address 129.0.0.0/8
set services nat rule r2 term t1 from source-address 128.0.0.0/8
set services nat rule r2 term t1 then translated source-pool p2
set services nat rule r2 term t1 then translated translation-type napt-44
set services nat rule r2 term t1 then translated address-pooling paired
set services nat rule r2 term t1 then syslog
set services stateful-firewall rule r2 match-direction input
set services stateful-firewall rule r2 term t1 from source-address any-unicast
set services stateful-firewall rule r2 term t1 then accept
set services stateful-firewall rule r2 term t1 then syslog
set services service-set ss2 replicate-services replication-threshold 180
set services service-set ss2 replicate-services stateful-firewall
set services service-set ss2 replicate-services nat
set services service-set ss2 stateful-firewall-rules r2
set services service-set ss2 nat-rules r2
set services service-set ss2 next-hop-service inside-service-interface vms-4/0/0.20
set services service-set ss2 next-hop-service outside-service-interface vms-4/0/0.30
set services service-set ss2 syslog host local class session-logs
set services service-set ss2 syslog host local class stateful-firewall-logs
set services service-set ss2 syslog host local class nat-logs

```

Configuring Interfaces for Chassis 1.

Step-by-Step Procedure

The interfaces for each of the HA pair of routers are configured identically with the exception of the following service PIC options:

- `redundancy-options redundancy-peer ipaddress address`
- `unit unit-number family inet address address of a unit, other than 0, that contains the ip-address-owner service-plane option`

To configure interfaces:

1. Configure the redundant service PIC on chassis 1.

```
[edit interfaces]
user@host# set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.2
user@host# set interfaces vms-4/0/0 redundancy-options routing-instance HA
user@host# set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
user@host# set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.1/32
user@host# set interfaces vms-4/0/0 unit 20 family inet
user@host# set interfaces vms-4/0/0 unit 20 service-domain inside
user@host# set interfaces vms-4/0/0 unit 30 family inet
user@host# set interfaces vms-4/0/0 unit 30 service-domain outside
```

2. Configure the interfaces for chassis 1 that are used as interchassis links for synchronization traffic.

```
user@host# set interfaces ge-2/0/0 vlan-tagging
user@host# set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.1/24
```

3. Configure remaining interfaces as needed.

Results

```
user@host# show interfaces
ge-2/0/0 {
    vlan-tagging;
    unit 0 {
        vlan-id 100;
        family inet {
            address 20.1.1.1/24;
        }
    }
}
vms-4/0/0 {
    redundancy-options {
        redundancy-peer {
            ipaddress 5.5.5.2;
        }
        routing-instance HA;
    }
    unit 10 {
        ip-address-owner service-plane;
    }
}
```

```

        family inet {
            address 5.5.5.1/32;
        }
    }
    unit 20 {
        family inet;
        family inet6;
        service-domain inside;
    }
    unit 30 {
        family inet;
        family inet6;
        service-domain outside;
    }
}

```

Configure Routing Information for Chassis 1

Step-by-Step Procedure

Detailed routing configuration is not included for this example. A routing instance is required for the HA synchronization traffic between the chassis as follows:

- Configure routing instances for Chassis 1.

```

user@host# set routing-instances HA instance-type vrf
user@host# set routing-instances HA interface ge-2/0/0.0
user@host# set routing-instances HA interface vms-4/0/0.10
user@host# set routing-instances HA route-distinguisher 1:1
user@host# set policy-options policy-statement dummy term 1 then reject
user@host# set routing-instances HA vrf-import dummy
user@host# set routing-instances HA vrf-export dummy
user@host# set routing-instances HA routing-options static route route 5.5.5.1/32 next-hop
vms-4/0/0.10
user@host# set routing-instances HA routing-options static route route 5.5.5.2/32 next-hop
20.1.1.2

```

Results

```

user@host# show routing-instances
HA {
    instance-type vrf;
    interface ge-2/0/0.0;
    interface vms-4/0/0.10;
    route-distinguisher 1:1;
    vrf-import dummy;
    vrf-export dummy;
    routing-options {
        static {
            route 5.5.5.1/32 next-hop vms-4/0/0.10;
            route 5.5.5.2/32 next-hop 20.1.1.2;
        }
    }
}

```

Configuring NAT and Stateful Firewall for Chassis 1

Step-by-Step Procedure

Configure NAT and stateful firewall identically on both routers. To configure NAT and stateful firewall:

1. Configure NAT as needed.

```

user@host# set services nat pool p2 address 32.0.0.0/24
user@host# set services nat pool p2 port automatic random-allocation
user@host# set services nat pool p2 address-allocation round-robin
user@host# set services nat rule r2 match-direction input
user@host# set services nat rule r2 term t1 from source-address 129.0.0.0/8
user@host# set services nat rule r2 term t1 from source-address 128.0.0.0/8
user@host# set services nat rule r2 term t1 then translated source-pool p2
user@host# set services nat rule r2 term t1 then translated translation-type napt-44
user@host# set services nat rule r2 term t1 then translated address-pooling paired
user@host# set services nat rule r2 term t1 then syslog

```

2. Configure stateful firewall as needed.

```

user@host# set services stateful-firewall rule r2 match-direction input
user@host# set services stateful-firewall rule r2 term t1 from source-address any-unicast
user@host# set services stateful-firewall rule r2 term t1 then accept
user@host# set services stateful-firewall rule r2 term t1 then syslog

```

Results

```

user@host# show services nat
nat {
    pool p2 {
        address 32.0.0.0/24;
        port {
            automatic {
                random-allocation;
            }
        }
        address-allocation round-robin;
    }
    rule r2 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    129.0.0.0/8;
                    128.0.0.0/8;
                }
            }
            then {
                translated {
                    source-pool p2;
                    translation-type {
                        napt-44;
                    }
                    address-pooling paired;
                }
                syslog;
            }
        }
    }
}

```

```

    }
  }
}

```

```

user@host show services stateful-firewall
rule r2 {
  match-direction input;
  term t1 {
    from {
      source-address {
        any-unicast;
      }
    }
    then {
      accept;
      syslog;
    }
  }
}
}

```

Configuring the Service Set

Step-by-Step Procedure

Configure the the service set identically on both routers. To configure the service set:

1. Configure the service set replication options.

```

user@host# set services service-set ss2 replicate-services replication-threshold 180
user@host# set services service-set ss2 replicate-services stateful-firewall
user@host# set services service-set ss2 replicate-services nat

```

2. Configure references to NAT and stateful firewall rules for the service set.

```

user@host# set services service-set ss2 stateful-firewall-rules r2
user@host# set services service-set ss2 nat-rules r2

```

3. Configure next-hop service interface on the vms-PIC.

```
user@host# set services service-set ss2 next-hop-service inside-service-interface vms-4/0/0.20
user@host# set services service-set ss2 next-hop-service outside-service-interface
vms-4/0/0.30
```

4. Configure desired logging options.

```
user@host# set services service-set ss2 syslog host local class session-logs
user@host# set services service-set ss2 syslog host local class stateful-firewall-logs
user@host# set services service-set ss2 syslog host local class nat-logs
```

Results

```
user@host# show services service-set ss2
syslog {
    host local {
        class {
            session-logs;
            inactive: stateful-firewall-logs;
            nat-logs;
        }
    }
}
replicate-services {
    replication-threshold 180;
    stateful-firewall;
    nat;
}
stateful-firewall-rules r2;
inactive: nat-rules r2;
next-hop-service {
    inside-service-interface vms-3/0/0.20;
    outside-service-interface vms-3/0/0.30;
}
}
```

Configuring Interfaces for Chassis 2

Step-by-Step Procedure

The interfaces for each of the HA pair of routers are configured identically with the exception of the following service PIC options:

- `redundancy-options redundancy-peer ipaddress address`
- `unit unit-number family inet address address` of a unit, other than 0, that contains the `ip-address-owner service-plane option`

1. Configure the redundant service PIC on chassis 2.

The `redundancy-peer ipaddress` points to the address of the unit (unit 10) on `vms-4/0/0` on chassis on chassis 1 that contains the `ip-address-owner service-plane` statement.

```
[edit interfaces]
set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.1
user@host# set interfaces vms-4/0/0 redundancy-options routing-instance HA
user@host# set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
user@host# set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.2/32
user@host# set interfaces vms-4/0/0 unit 20 family inet
user@host# set interfaces vms-4/0/0 unit 20 service-domain inside
user@host# set interfaces vms-4/0/0 unit 30 family inet
user@host# set interfaces vms-4/0/0 unit 30 service-domain outside
```

2. Configure the interfaces for chassis 2 that are used as interchassis links for synchronization traffic

```
user@host# set interfaces ge-2/0/0 vlan-tagging
user@host# set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.2/24
```

3. Configure remaining interfaces for chassis 2 as needed.

Results

```
user@host# show interfaces
vms-4/0/0 {
    redundancy-options {
        redundancy-peer {
            ipaddress 5.5.5.1;
```

```

    }
    routing-instance HA;
}
unit 0 {
    family inet;
}
unit 10 {
    ip-address-owner service-plane;
    family inet {
        address 5.5.5.2/32;
    }
}
}
ge-2/0/0 {
    vlan-tagging;
    unit 0 {
        vlan-id 100;
        family inet {
            address 20.1.1.2/24;
        }
    }
    unit 10 {
        vlan-id 10;
        family inet {
            address 2.10.1.2/24;
        }
    }
}

```

Configure Routing Information for Chassis 2

Step-by-Step Procedure

Detailed routing configuration is not included for this example. A routing instance is required for the HA synchronization traffic between the two chassis and is included here.

- Configure routing instances for chassis 2.

```

user@host# set routing-instances HA instance-type vrf
user@host# set routing-instances HA interface ge-2/0/0.0
user@host# set routing-instances HA interface vms-4/0/0.10
user@host# set routing-instances HA route-distinguisher 1:1
user@host# set policy-options policy-statement dummy term 1 then reject
user@host# set routing-instances HA vrf-import dummy

```



```

user@host# set routing-instances HA vrf-export dummy
user@host# set routing-instances HA routing-options static route 5.5.5.2/32 next-hop
vms-4/0/0.10
user@host# set routing-instances HA routing-options static route 5.5.5.1/32 next-hop 20.1.1.1

```



NOTE: The following configuration steps are *identical* to the steps shown for chassis 1.

- Configuring NAT and Stateful Firewall
- Configuring the Service Set

Results

```

user@host# show services routing-instances
HA {
    instance-type vrf;
    interface xe-2/2/0.0;
    interface vms-4/0/0.10;
    route-distinguisher 1:1;
    vrf-import dummy;
    vrf-export dummy;
    routing-options {
        static {
            route 5.5.5.2/32 next-hop vms-4/0/0.10;
            route 5.5.5.1/32 next-hop 20.1.1.1;
        }
    }
}

```

Example: Configuring AutoVPN with Pre-Shared Key

IN THIS SECTION

- [Requirements | 101](#)
- [Configure different IKE preshared key | 101](#)
- [Configure same IKE preshared key | 103](#)

This example shows how to configure different IKE preshared key used by the VPN gateway to authenticate the remote peer. Similarly, to configure same IKE preshared key used by the VPN gateway to authenticate the remote peer.

Refer other examples in this topic for end-to-end configuration of AutoVPN.

Requirements

This example uses the following hardware and software components:

- MX240, MX480, and MX960 with MX-SPC3 and Junos OS Release 21.1R1 that support AutoVPN
- or SRX5000 line with SPC3 and Junos OS Release 21.2R1 that support AutoVPN
- or vSRX Virtual Firewall running ike process (with the `junos-ike` package) and Junos OS Release 21.2R1 that support AutoVPN

Configure different IKE preshared key

To configure different IKE preshared key that the VPN gateway uses to authenticate the remote peer, perform these tasks.

1. Configure the seeded preshared for IKE policy in the device with AutoVPN hub.

[edit]

```
user@host# set security ike policy IKE_POL seeded-pre-shared-key ascii-text ascii-text
```

or

```
user@host# set security ike policy IKE_POL seeded-pre-shared-key hexadecimal hexadecimal
```

For example:

```
user@host# set security ike policy IKE_POL seeded-pre-shared-key ascii-text
ThisIsMySecretPreSharedkey
```

or

```
user@host# set security ike policy IKE_POL seeded-pre-shared-key hexadecimal
5468697349734d79536563726563745072655368617265646b6579
```

2. Display the pre-shared key for remote peer using gateway name and user-id.

```
[edit]
user@host> show security ike pre-shared-key gateway gateway-name user-id user-id
```

For example:

```
user@host> show security ike pre-shared-key gateway-name HUB_GW user-id user1@juniper.net
```

Pre-shared key: 79e4ea39f5c06834a3c4c031e37c6de24d46798a

3. Configure the generated PSK ("79e4ea39f5c06834a3c4c031e37c6de24d46798a" in ["step 2" on page 102](#)) in the ike policy on the remote peer device.

```
[edit]
user@peer# set security ike policy IKE_POL pre-shared-key ascii-text generated-psk
```

For example:

```
user@peer# set security ike policy IKE_POL pre-shared-key ascii-text
79e4ea39f5c06834a3c4c031e37c6de24d46798a
```

4. (Optional) To bypass the IKE ID validation and allow all IKE ID types, configure `general-ikeid` configuration statement under the `[edit security ike gateway gateway_name dynamic]` hierarchy level in the gateway.

```
[edit]
user@host# set security ike gateway HUB_GW dynamic general-ikeid
```

Result

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

```
[edit]
user@host> show security
  ike {
    proposal IKE_PROP {
```

```

        authentication-method pre-shared-keys;
        dh-group group14;
        authentication-algorithm sha-256;
        encryption-algorithm aes-256-cbc;
        lifetime-seconds 750;
    }
    policy IKE_POL {
        proposals IKE_PROP;
        seeded-pre-shared-key ascii-text "$9$zoDln9pIEyWLN0BLNdboaFn/C0BRhSeM8"; ##SECRET-DATA
    }
    gateway HUB_GW {
        ike-policy IKE_POL;
        dynamic {
            general-ikeid;
            ike-user-type group-ike-id;
        }
        local-identity hostname hub.juniper.net;
        external-interface lo0.0;
        local-address 11.0.0.1;
        version v2-only;
    }
}

```

Configure same IKE preshared key

To configure same IKE preshared key that the VPN gateway uses to authenticate the remote peer, perform these tasks.

1. Configure the common pre-shared-key for ike policy in the device with AutoVPN hub.

```

[edit]
user@host# set security ike policy IKE_POL pre-shared-key ascii-text ascii text

```

For example:

```

user@host# # set security ike policy IKE_POL pre-shared-key ascii-text
ThisIsMySecretPreSharedkey

```

2. Configure the common pre-shared-key on the ike policy for remote peer device.

```
[edit]
user@peer# set security ike policy IKE_POL pre-shared-key ascii-text ascii text
```

For example:

```
user@peer# set security ike policy IKE_POL pre-shared-key ascii-text
ThisIsMySecretPreSharedkey
```

3. (Optional) To bypass the IKE ID validation and allow all IKE ID types, configure general-ikeid configuration statement under the [edit security ike gateway *gateway_name* dynamic] hierarchy level in the gateway.

```
[edit]
user@host# set security ike gateway HUB_GW dynamic general-ikeid
```

Result

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

```
[edit]
user@host> show security
  ike {
    proposal IKE_PROP {
      authentication-method pre-shared-keys;
      dh-group group14;
      authentication-algorithm sha-256;
      encryption-algorithm aes-256-cbc;
      lifetime-seconds 750;
    }
    policy IKE_POL {
      proposals IKE_PROP;
      pre-shared-key ascii-text "$9$wo2oGk.569pDi9p0BSys24"; ## SECRET-DATA
    }
    gateway HUB_GW {
      ike-policy IKE_POL;
      dynamic {
```

```

        general-ikeid;
        ike-user-type group-ike-id;
    }
    local-identity user-at-hostname user1@juniper.net;
    external-interface lo0;
    local-address 11.0.0.1;
    version v2-only;
}
}

```

Enabling and Disabling Next Gen Services

IN THIS SECTION

- [Loading the Software Images on Next-Generation Routing Engines | 105](#)
- [Enabling Next Gen Services on an MX Series Router | 107](#)
- [Disabling Next Gen Services on an MX Series Router | 108](#)
- [Determining Whether Next Gen Services is Enabled on an MX Series Router | 109](#)

To use Next Gen Services, you must first enable it on the MX Series router. This topic describes how to enable Next Gen Services, how to disable Next Gen Services, and how to determine whether Next Gen Services is enabled or disabled on your system.

Loading the Software Images on Next-Generation Routing Engines

The Next-Gen Services MX-SPC3 services card can exhibit inconsistent behavior when the vmhost image is installed on the Next-Generation Routing Engines listed:

1. RE-S-X6-64G-BB (NG-RE)
2. RE-S-X6-64G-UB (NG-RE)
3. RE-S-X6-64G-S (NG-RE)
4. RE-S-X6-64G-R (NG-RE)
5. RE-S-X6-128G-S-BB (NG-RE)

6. RE-S-X6-128G-S-S (NG-RE)

7. RE-S-X6-128G-S-R (NG-RE)

You can use `show chassis hardware | match Routing` or `show chassis hardware models | match Routing` command to show the model of the Routing Engines.

For example:

```
user@host# show chassis hardware | match Routing
Routing Engine 0 REV 19 750-054758 CAPS2198 RE-S-2X00x6
Routing Engine 1 REV 19 750-054758 CAPT1717 RE-S-2X00x6
```

or

```
user@host# show chassis hardware models | match Routing
Routing Engine 0 REV 19 750-054758 CAPS2198 RE-S-X6-64G-S
Routing Engine 1 REV 19 750-054758 CAPT1717 RE-S-X6-64G-S
```

This behavior can result in you encountering one of the following:

- The MX-SPC3 card remains in Present state and does not come online
- The MX-SPC3 comes online successfully with different a software image (either a previously installed image or the pre-loaded image from manufacturing)

To work around this problem, you must install the **jpfe-spc3*** package manually on the NG-RE. To install this package manually, follow one of these procedures, depending on whether or not you have enabled Next Gen Services (unified-services) mode:

If unified-services are enabled:

1. Download the **jpfe-spc3*** required package for the Junos "New Gen Services for 32 Bit-MX High-End Series" **vmhost** onto the routing engine (RE) from [Downloads](#). For example, to download the **jpfe-spc3*** package from the [MX960 Downloads](#) page, select the required Junos OS version, then select the matching "New Gen Services for 32 Bit-MX High-End Series" entry and load it onto the RE. Unified services must be enabled on all routing engines on the device.
2. Load the selected **vmhost*** image on the RE.
3. After the RE boots, copy the **jpfe-spc3*** package to the **/var/tmp** directory
4. Load the **jpfe-spc3*** package. Modify the command to match your specific **jpfe-spc3*** version:

```
user@host> request system software add /var/tmp/jpfe-spc3-mx-x86-32-19.4R1.9.tgz reboot
```

If unified-services are disabled:

1. Download the **jpfe-spc3*** required package for the Junos "New Gen Services for 32 Bit-MX High-End Series" **vmhost** onto the routing engine (RE) from [Downloads](#). For example, to download the **jpfe-spc3*** package from the [MX960 Downloads](#) page, select the required Junos OS version, then select the matching "New Gen Services for 32 Bit-MX High-End Series" entry and load it onto the RE.
2. Load the desired **vmhost*** image on the RE
3. After the RE boots, enable unified-services mode:

```
user@host> request system enable unified-services
```

4. Copy package **jpfe-spc3*** package to the **/var/tmp** directory.
5. Load the **jpfe-spc3*** package. Modify the command to match your specific **jpfe-spc3*** version:

```
user@host> request system software add /var/tmp/jpfe-spc3-mx-x86-32-19.4R1.9.tgz reboot
```



NOTE: When MX-SPC3 card is installed on an MX chassis, misconfig alarm is reported with the reason as FPC in unsupported mode. This alarm might be seen when the unified services is disabled.



NOTE: In systems with two Next-Generation Routing Engines with primary as (**re0**) and the backup as (**re1**), both Next-Generation Routing Engines cannot be configured to be primary at the same time. This configuration causes the commit check to fail.

Enabling Next Gen Services on an MX Series Router

There are specific steps you'll need to take if you're migrating your services from MS-MPC cards to the MX-SPC3 services cards. The Next Gen Services CLI differs from these legacy services.

The following procedure is a general procedure for enabling and disabling Next Gen Services.

Before you do anything, you'll need to back up your configuration.

For more details on the differences between the configuration of the MX-SPC3 services card and legacy services cards, see ["Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3"](#) on page 16 and plan your migration appropriately.

You can run Next Gen Services on the MX240, MX480 and MX960 using the MX-SPC3 services card. To use Next Gen Services on the MX Series, you must first enable Next Gen Services:

1. Delete any router configuration that is for services. This includes configuration under the [edit services] hierarchy, configuration for services interfaces, and any configuration that refers to services interfaces.
2. Enable Next Gen Services.

```
user@host> request system enable unified-services
```

3. When the following message appears, enter **yes**.

```
Before enabling unified services, please move to baseline configuration.
Are above conditions satisfied ? [yes,no]
```

4. Reboot the MX Series chassis.

```
user@host> request system reboot
```

You can also enable the Next Gen Services on a Guest network function (GNF), by using the CLI `request system enable unified-services` at the GNF level. For more information, see *Next Gen Services on Junos node slicing*.

Disabling Next Gen Services on an MX Series Router

To disable Next Gen Services on the MX Series:

1. Delete any router configuration that is for services. This includes configuration under the [edit services] hierarchy, configuration for services interfaces, and any configuration that refers to services interfaces.
2. Disable Next Gen Services.

```
user@host> request system disable unified-services
```

3. When the following message appears, enter **yes**.

```
Before disabling unified services, please move to baseline configuration.
Are above conditions satisfied ? [yes,no]
```

```
Unified-Services downgrade staged. Please reboot with 'request system reboot' command to
complete the downgrade
```

```
WARNING: cli has been replaced by an updated version:
CLI release 20190829.221548_builder.r1052644 built by builder on 2019-08-29 22:27:13 UTC
Restart cli using the new version ? [yes,no] (yes)
```

4. Reboot the MX Series chassis.

```
user@host> request system reboot
```

Determining Whether Next Gen Services is Enabled on an MX Series Router

To determine whether Next Gen Services is enabled:

- Enter the following command:

```
user@host> show system unified-services status
```

One of the following messages appears:

- Enabled—Next Gen Services is enabled and ready to use.
- Unified Services : Upgrade staged , please reboot with 'request system reboot' to enable unified services.
—You must perform a system reboot before Next Gen Services is enabled.
- Disabled—Next Gen Services is disabled.
- Unified Services : Upgrade staged , please reboot with 'request system reboot' to disable unified services.
—You must perform a system reboot before Next Gen Services is disabled.

RELATED DOCUMENTATION

[Next Gen Services Overview | 2](#)

[Next Gen Services Feature Configuration Overview | 79](#)

[Configuration Differences Between Adaptive Services and Next Gen Services on the MX-SPC3 | 16](#)

Global System Logging Overview and Configuration

IN THIS CHAPTER

- [Understanding Next Gen Services CGNAT Global System Logging | 110](#)
- [Enabling Global System Logging for Next Gen Services | 112](#)
- [Configuring Local System Logging for Next Gen Services | 113](#)
- [Configuring System Logging to One or More Remote Servers for Next Gen Services | 115](#)
- [System Log Error Messages for Next Gen Services | 118](#)
- [Configuring Syslog Events for NAT Rule Conditions with Next Gen Services | 127](#)

Understanding Next Gen Services CGNAT Global System Logging

IN THIS SECTION

- [Next Gen Services CGNAT Global System Logging | 110](#)
- [Modes of Operation for Next Gen Services System Logging | 111](#)
- [Understanding Stream Mode | 111](#)
- [System Logging Configuration Overview | 111](#)
- [Disabling Session Open Information in Syslogs | 112](#)

All CGNAT services supported under Next Gen Services use global system logging. This topic describes global system logging for Next Gen Services CGNAT services and how to configure it.

Next Gen Services CGNAT Global System Logging

The CGNAT services supported under Next Gen Services support global system logging for syslog messages. You configure syslog messaging for these services under the service-set hierarchy. You can

send logs to either the local routing engine (RE) or one or more remote servers (each of these is identified as a stream). You can configure files to log system messages and also assign attributes, such as severity levels, to messages. Reboot requests are recorded to the system log files, which you can view with the `show log` command.

In the case of an AMS bundle, each PIC establishes a TCP connection with the log server and the external collector receives log messages from all the AMS members.

Modes of Operation for Next Gen Services System Logging

You can save logs for Next Gen Services locally, which is called: event mode, or send the log messages to one or more external servers, called: stream mode.

In event mode, after the log message is recorded, the log is stored within a log file which is then stored in the database table of the local routing engine (RE) for further analysis.

When configured in stream mode, log messages are streamed to one or more remote log servers. Each remote log server is assigned a stream from which it receives logs.

Understanding Stream Mode

When configured in stream mode, Next Gen Services log messages are streamed to a remote device.

For stream mode log forwarding, you can configure which transport protocol is used between MX-SPC3 services card and the log server. You can use either UDP, TCP, or TLS as the transport protocol.

When the device is configured in stream mode, you can configure a maximum of eight system log hosts to stream to.

System Logging Configuration Overview

Configuring system logging for Next Gen Services involves several main steps and considerations:

- Global system logging — Next Gen Services system logging uses a global logging option that you need to enable in order to collect system log messages.

To enable global system logging for Next Gen Services, set the `traceoptions` option under the `edit services rtlog` hierarchy.

- For Next Gen Services, syslogs are always set at the service-set level regardless of whether you are running event mode or stream mode.

You must configure system logging for each service-set for which you want to collect logs. Each service-set uses a separate TCP connection in stream mode.

As a log client, Next Gen Services initiates TCP/TLS connections to the remote log server. By default, we connect to port 514 for TCP logging [RFC 6587], and port 6514 for TLS logging [RFC 5425]. You can also specify port numbers for TCP and TLS logging using CLI.

- If you are using AMS bundles, syslogs are generated from each member interface of AMS group

Disabling Session Open Information in Syslogs

You can stop open session information from cluttering up your syslogs by disabling session open information from being collected:

```
user@host# set services service-set ss1 service-set-options disable-session-open-syslog
```

RELATED DOCUMENTATION

[Enabling Global System Logging for Next Gen Services | 112](#)

[Configuring System Logging to One or More Remote Servers for Next Gen Services | 115](#)

[Configuring Local System Logging for Next Gen Services | 113](#)

Enabling Global System Logging for Next Gen Services

To configure either event mode or stream mode system logging for Next Gen Services, you must first globally enable logging:

1. Enable system logging for Next Gen Services.

```
[edit]
user@host# edit services rtlogtraceoptions
```

2. Enable unified-services on all routing engines on the device.

```
[edit]
user@host# request system enable unified-services
```

3. Specify the groups from which to inherit configuration data.

```
[edit services rtlog traceoptions]
user@host# set apply-groups group-names
```

4. Specify which groups not to inherit configuration data from.

```
[edit services rtlog traceoptions]
user@host# set apply-groups-except group-names
```

5. Configure information about the files that contain trace logging information.

```
[edit services rtlog traceoptions]
user@host# set file filename
```

6. Define tracing operations for individual service-sets. To specify more than one tracing operation, include multiple flag statements.

```
[edit services rtlog traceoptions]
user@host# set flag flag, flag...
```

7. (Optional) If you prefer not to perform any system logging, you can disable it.

```
[edit services rtlog traceoptions]
user@host# set no-remote-trace
```

RELATED DOCUMENTATION

[Understanding Next Gen Services CGNAT Global System Logging | 110](#)

[Configuring System Logging to One or More Remote Servers for Next Gen Services | 115](#)

[Configuring Local System Logging for Next Gen Services | 113](#)

Configuring Local System Logging for Next Gen Services

You must enable global system logging for Next Gen Services in order to perform event mode system logging. See, "[Enabling Global System Logging for Next Gen Services](#)" on page 112.

To send Next Gen Services log messages to a file on the local router, you'll need to configure system logging for event mode. This procedure describes this configuration process.



NOTE: For Next Gen Services, syslogs are always set at the service-set level. You must perform this procedure for each service-set for which you want to collect logs.

To configure event mode logging for Next Gen Services:

1. Specify the filename to send log messages to.

```
user@host# set system syslog file filename
```

2. Specify the name of the service-set for which you want to log messages.

```
user@host# edit services service-set service-set-name syslog
```

For example specify the service-set name to ss1.

```
user@host# edit services service-set ss1 syslog
```

3. Specify the security transport protocol for syslog messages.

```
[edit services service-set ss1 syslog]
user@host# set transport protocol tls | tcp | udp
```

4. Enable event mode system logging for the service-set.

```
[edit services service-set ss1 syslog]
user@host# set mode event
```

5. Specify the rate at which log messages are sent per second.

```
[edit services service-set ss1 syslog]
user@host# set event-rate 100
```

6. Specify a local tag name for the log messages.

```
[edit services service-set ssl syslog]
user@host# set local-log-tag SYSLOG
```

7. Specify the categories for which you want to collect events.

```
[edit services service-set ssl syslog]
user@host# set local-category category, category
```

For example, to collect logs for stateful firewall, sessions and NAT:

```
[edit services service-set ssl syslog]
user@host# set local-category sfw, session, nat
```



NOTE: local-log-tag and local

RELATED DOCUMENTATION

[Enabling Global System Logging for Next Gen Services | 112](#)

[Understanding Next Gen Services CGNAT Global System Logging | 110](#)

[Configuring System Logging to One or More Remote Servers for Next Gen Services | 115](#)

Configuring System Logging to One or More Remote Servers for Next Gen Services

You must enable global system logging for Next Gen Services in order to perform stream logging. See, ["Enabling Global System Logging for Next Gen Services" on page 112](#).

To send system log messages about Next Gen Services to one or more remote servers, you can configure system logging for stream mode. This procedure describes the configuration process.



NOTE: Next Gen Services system log messages are configured and collected at the service-set level.

In this procedure, you'll configure a stream for the log messages between each service set and each remote server that you want to send log messages.

Complete this procedure for each service-set and each remote server for which you want to collect logs and send logs.

To configure stream mode system logging for Next Gen Services:

1. Specify the names of the service-set for which you want to collect log messages.

```
user@host# edit services service-set service-set-name syslog
```

For example specify the service-set name to ss1.

```
user@host# edit services service-set ss1 syslog
```

2. Specify the security transport protocol for syslog messages.

```
[edit services service-set ss1 syslog]
user@host# set transport protocol tls | tcp | udp
```

3. (Optional) Specify the syslog source address.

```
[edit services service-set ss1 syslog]
user@host# set source-address 50.0.0.10
```



BEST PRACTICE: The syslog source address can be any arbitrary IP address. It does not have to be an IP address that is assigned to the device. Rather, this IP address is used on the syslog collector to identify the syslog source. The best practice is to configure the source address as the IP address of the interface that the traffic is sent out on.

4. Specify a local tag name for the log messages.

```
[edit services service-set ssl syslog]
user@host# set local-log-tag SYSLOG
```

5. Enable stream mode system logging for the service-set.

```
[edit services service-set ssl syslog]
user@host# set modestream
```

6. Specify a name for the stream.

```
[edit services service-set ssl syslog]
user@host# set stream stream-name
```

For example, let's call the stream: stream-aa

```
[edit services service-set ssl syslog]
user@host# edit stream stream-aa
```

7. Specify the categories for which you want to collect events.

```
[edit services service-set ssl syslog stream stream-aa]
user@host# set category
```

For example, to collect logs for stateful firewall, sessions and NAT:

```
[edit services service-set ssl syslog stream stream-aa]
user@host# set category sfw, session, nat
```

8. Specify the file format for the log.

```
[edit services service-set ssl syslog stream stream-aa]
user@host# set format sd-syslog
```

9. Specify the IP address of syslog server to receive log messages,

```
[edit services service-set ssl syslog stream stream-aa]
user@host# set host address
```

10. Specify the level of severity for the stream.

```
[edit services service-set ssl syslog stream stream-aa]
user@host# set severity severity-level
```

RELATED DOCUMENTATION

[Understanding Next Gen Services CGNAT Global System Logging | 110](#)

[Enabling Global System Logging for Next Gen Services | 112](#)

[Configuring Local System Logging for Next Gen Services | 113](#)

System Log Error Messages for Next Gen Services

IN THIS SECTION

- [Session Open Logs | 119](#)
- [Session Close Logs | 120](#)
- [NAT Out of Address Logs | 121](#)
- [NAT Out of Ports Logs | 122](#)
- [NAT Rule Match Logs | 122](#)
- [NAT Pool Release Logs | 122](#)
- [NAT Port Block Allocation Logs | 123](#)
- [NAT Port Block Allocation Interim Logs | 123](#)
- [NAT Port Block Release Logs | 124](#)
- [Deterministic NAT Logs | 124](#)
- [Stateful Firewall Rule Accept Logs | 124](#)
- [Stateful Firewall Rule Reject Logs | 125](#)

- [Stateful Firewall Rule Discard Logs | 125](#)
- [Stateful Firewall Rule No Rule Drop Logs | 126](#)
- [Stateful Firewall No Policy Drop Logs | 126](#)

This topic describes Next Gen Services MX-SPC3 services card system log error messages and provides a comparison of these messages with the MS-MPC services card.

Session Open Logs

Following are example session open logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
JSERVICES_SESSION_OPEN application source-interface-name source-address source-port source-nat-information
destination-address destination-port destination-nat-information protocol-name software-information;
```

MX-SPC3 Services Card

```
RT_FLOW_SESSION_CREATE_USF Prefix service-set-name source-interface-name source-address source-port destination-
address destination-port service-name nat-source-address nat-source-port nat-destination-address nat-destination-
port src-nat-rule-type src-nat-rule-name dst-nat-rule-type dst-nat-rule-name protocol-name policy-name application
software-information;
```

Sample MX-SPC3 Output

A sample output is as follows:

```
<14>1 2018-06-26T17:23:06.269-07:00 book1et RT_FLOW - RT_FLOW_SESSION_CREATE_USF [junos@2636.1.1.1.2.25
prefix="SYSLOG-PREFIX" service-set-name="JNPR-NH-SSET3" source-address="50.0.0.10" source-port="1" destination-
address="60.0.0.10" destination-port="21219" connection-tag="0" service-name="icmp" nat-source-address="100.0.0.1"
nat-source-port="1024" nat-destination-address="60.0.0.10" nat-destination-port="21219" nat-connection-tag="0"
src-nat-rule-type="source rule" src-nat-rule-name="SRC-NAT-RULE1" dst-nat-rule-type="N/A" dst-nat-rule-name="N/A"
protocol-id="1" policy-name="p1" source-zone-name="JNPR-NH-SSET3-ZoneIn" destination-zone-name="JNPR-NH-SSET3-
ZoneOut" session-id-32="160000001" username="N/A" roles="N/A" packet-incoming-interface="vms-2/0/0.100"
application="UNKNOWN" nestedapplication="UNKNOWN" encrypted="UNKNOWN" application-category="N/A" application-sub-
category="N/A" application-risk="-1"] Prefix PADDY3 svc-set-name JNPR-NH-SSET3: session created 50.0.0.10/1-
>60.0.0.10/21219 0x0 icmp 100.0.0.1/1024->60.0.0.10/21219 0x0 source rule SRC-NAT-RULE1 N/A N/A 1 p1 JNPR-NH-
SSET3-ZoneIn JNPR-NH-SSET3-ZoneOut 160000001 N/A(N/A) vms-2/0/0.100 UNKNOWN UNKNOWN UNKNOWN N/A N/A -1
```

Session Open Logs With NAT

MS-MPC Services Card

```
SYSLOG_MSMP{SS_TEST}JSERVICES_SESSION_OPEN: application:ike-esp-nat, xe-2/2/1.0 24.0.0.2:1234 [85.0.0.1:1024] ->
25.0.0.2:1234 (UDP)
```

MX-SPC3 Services Card

```
Aug 3 02:04:28 mobst480i RT_FLOW: RT_FLOW_SESSION_CREATE_USF: Tag svc-set-name sset1: session created 90.0.0.2/1-
>30.0.0.2/4323 0x0 icmp 50.0.0.3/1024->30.0.0.2/4323 0x0 source rule rule1 N/A N/A 1 p1 sset1-ZoneIn sset1-ZoneOut
160000015 N/A(N/A) vms-2/0/0.1 UNKNOWN UNKNOWN UNKNOWN N/A N/A -1 N/A
```

Session Open Logs Without NAT

MS-MPC Services Card

```
SYSLOG_MSMP{SS_TEST}JSERVICES_SESSION_OPEN: application:ike-esp-nat, xe-2/2/1.0 24.0.0.2:1234 -> 25.0.0.2:1234
(UDP)
```

MX-SPC3 Services Card

```
RT_FLOW - RT_FLOW_SESSION_CREATE_USF [junos@2636.1.1.1.2.25 tag="SYSLOG_SFW" service-set-name="ss1" source-
address="20.1.1.2" source-port="12000" destination-address="30.1.1.2" destination-port="22000" connection-tag="0"
service-name="None" nat-source-address="20.1.1.2" nat-source-port="12000" nat-destination-address="30.1.1.2" nat-
destination-port="22000" nat-connection-tag="0" src-nat-rule-type="N/A" src-nat-rule-name="N/A" dst-nat-rule-
type="N/A" dst-nat-rule-name="N/A" protocol-id="6" policy-name="policy1" source-zone-name="ss1-ZoneIn"
destination-zone-name="ss1-ZoneOut" session-id-32="190000004" username="N/A" roles="N/A" packet-incoming-
interface="xe-5/3/2.0" application="UNKNOWN" nested-application="UNKNOWN" encrypted="UNKNOWN" application-
category="N/A" application-sub-category="N/A" application-risk="-1" application-characteristics="N/A"] Tag
SYSLOG_SFW svc-set-name ss1: session created 20.1.1.2/12000->30.1.1.2/22000 0x0 None 20.1.1.2/12000-
>30.1.1.2/22000 0x0 N/A N/A N/A N/A 6 policy1 ss1-ZoneIn ss1-ZoneOut 190000004 N/A(N/A) xe-5/3/2.0 UNKNOWN UNKNOWN
UNKNOWN N/A N/A -1 N/A
```

Session Close Logs

Following are example session close logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
JSERVICES_SESSION_CLOSE application source-interface-name source-address source-port source-nat-information
destination-address destination-port destination-nat-information protocol-name software-information;
```

MX-SPC3 Services Card

```
RT_FLOW_SESSION_CLOSE_USF Prefix service-set-name source-interface-name source-address source-port destination-
address destination-port service-name nat-source-address nat-source-port nat-destination-address nat-destination-
port src-nat-rule-type src-nat-rule-name dst-nat-rule-type dst-nat-rule-name protocol-name policy-name; software-
information;
```

Sample MX-SPC3 Output

A sample output follows:

```
<14>1 2018-06-27T09:24:00.058-07:00 booklet RT_FLOW - RT_FLOW_SESSION_CLOSE_USF [junos@2636.1.1.1.2.25
prefix="SYSLOG-PREFIX" service-set-name="JNPR-NH-SSET3" reason="idle Timeout" source-address="50.0.0.10" source-
port="1" destination-address="60.0.0.10" destination-port="30170" connection-tag="0" service-name="icmp" nat-
source-address="100.0.0.1" nat-source-port="1024" nat-destination-address="60.0.0.10" nat-destination-port="30170"
nat-connection-tag="0" src-nat-rule-type="source rule" src-nat-rule-name="SRC-NAT-RULE1" dst-nat-rule-type="N/A"
dst-nat-rule-name="N/A" protocol-id="1" policy-name="p1" source-zone-name="JNPR-NH-SSET3-ZoneIn" destination-zone-
name="JNPR-NH-SSET3-ZoneOut" session-id-32="160000001" packets-from-client="1" bytes-from-client="84" packets-
from-server="0" bytes-from-server="0" elapsed-time="4" application="UNKNOWN" nested-application="UNKNOWN"
username="N/A" roles="N/A" packet-incoming-interface="vms-2/0/0.100" encrypted="UNKNOWN" application-
category="N/A" application-sub-category="N/A" application-risk="-1"] Prefix PADDY-DEF svc-set-name JNPR-NH-SSET3:
session closed idle Timeout: 50.0.0.10/1->60.0.0.10/30170 0x0 icmp 100.0.0.1/1024->60.0.0.10/30170 0x0 source rule
SRC-NAT-RULE1 N/A N/A 1 p1 JNPR-NH-SSET3-ZoneIn JNPR-NH-SSET3-ZoneOut 160000001 1(84) 0(0) 4 UNKNOWN UNKNOWN
N/A(N/A) vms-2/0/0.100 UNKNOWN N/A N/A -1
```

NAT Out of Address Logs

Following are example NAT Out of Address logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
JSERVICES_NAT_OUTOF_ADDRESSES: nat-pool-name
```

MX-SPC3 Services Card:

```
Aug 10 10:06:13 champ RT_NAT: RT_SRC_NAT_OUTOF_ADDRESSES: nat-pool-name src_pool1 is out of addresses
```

NAT Out of Ports Logs

Following are example NAT Out of Ports logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
{NPU-1-PFX1}[jservices-nat]: JSERVICES_NAT_OUTOF_PORTS: natpool NAT-POOL-NPU1-PFX3 is out of ports
```

MX-SPC3 Services Card

```
jul 31 03:08:30 esst480h RT_NAT: RT_SRC_NAT_OUTOF_PORTS: nat-pool-name nat_pool1 is out of ports
```

NAT Rule Match Logs

Following are example NAT rule match logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
SYSLOG_MSMP{SS_TEST}[jservices-nat]: JSERVICES_NAT_RULE_MATCH: proto 17 (UDP) application: any,  
xe-2/2/1.0:24.0.0.2:1234 -> 25.0.0.2:1234, Match NAT rule-set: (null), rule: NAT_RULE_TEST, term: t
```

MX-SPC3 Services Card

```
RT_NAT: RT_NAT_RULE_MATCH: protocol-id 17 protocol-name udp application Unknown interface-name ge-2/0/9.0 source-  
address 11.1.1.2 source-port 2000 destination-address 12.1.1.2 destination-port 5000 rule-set-name rule-set rule-  
name nat-rule
```

NAT Pool Release Logs

Following are example NAT Rule Match logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
SYSLOG_MSMP{SS_TEST}[jservices-nat]: JSERVICES_NAT_POOL_RELEASE: natpool release 85.0.0.1:1024[1]
```

MX-SPC3 Services Card

```
RT_NAT: RT_SRC_NAT_POOL_RELEASE: nat-pool-name nat-pool address 112.1.1.4 port 1024 count 1
```

NAT Port Block Allocation Logs

Following are example NAT port block allocation logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card-Example 1

```
SYSLOG_MSMP{ss1}[jservices-nat]: JSERVICES_NAT_PORT_BLOCK_ALLOC: 11.1.1.2 -> 112.1.1.4:42494-42503 0x59412760
```

MX-SPC3 Services Card-Example 1

```
Aug 9 23:01:59 esst480r RT_NAT: RT_SRC_NAT_PBA_ALLOC: Subscriber 20.1.1.5 used/maximum [1/1] blocks, allocates port block [49774-49923] from 100.0.0.1 in source pool p1 lsys_id: 0
```

MS-MPC Services Card-Example 2

```
SYSLOG_MSMP{ss1}[jservices-nat]: JSERVICES_NAT_PORT_BLOCK_RELEASE: 2001:2010:0:0:0:0:2 -> 161.161.16.1:56804-56813 0x597ef2c3
```

MX-SPC3 Services Card-Example 2

```
RT_NAT: RT_SRC_NAT_PBA_ALLOC: Subscriber 11.1.1.2 used/maximum [1/2] blocks, allocates port block [13934-13943] from 112.1.1.1 in source pool nat-pool lsys_id: 0
```

NAT Port Block Allocation Interim Logs

Following are example interim logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
SYSLOG_MSMP{ss1}[jservices-nat]: JSERVICES_NAT_PORT_BLOCK_ACTIVE: 11.1.1.2 -> 112.1.1.4:42494-42503 0x59412760
```

MX-SPC3 Services Card

```
RT_NAT: RT_SRC_NAT_PBA_INTERIM: Subscriber 50.0.0.3 used/maximum [1/1] blocks, allocates port block [5888-6015] from 202.0.0.1 in source pool JNPR-CGNAT-PUB-POOL lsys_id: 0
```


NAT Port Block Release Logs

Following are example NAT port block release logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
JSERVICES_NAT_PORT_BLOCK_RELEASE source-address nat-source-address nat-source-port-range-start nat-source-port-range-end object-create-time;
```

MX-SPC3 Services Card

```
RT_NAT: RT_SRC_NAT_PBA_RELEASE: Subscriber 11.1.1.2 used/maximum [2/3] blocks, releases port block [3839-3843] from 112.1.2.1 in source pool nat-pool lsys_id: 0
```

Deterministic NAT Logs

MS-MPC Services Card

```
{ss1}[jservices-nat]: JSERVICES_DET_NAT_CONFIG: Deterministic NAT Config [2001:2010::-2001:2010::ff]: [161.161.16.1-161.161.16.254]:0:200:0:1024-65535
```

Stateful Firewall Rule Accept Logs

Following are example stateful firewall rule accept logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
Sep 20 01:36:51 mobst480b (FPC Slot 5, PIC Slot 0) 2017-09-20 08:36:19: SYSLOG_MSMP{SS_TEST}[jservices-sfw]: JSERVICES_SFW_RULE_ACCEPT: proto 17 (UDP) application: any, interface: xe-2/2/1.0, 24.0.0.2:1234 -> 25.0.0.2:1234, Match SFW allow rule-set: (null), rule: SFW_RULE_TEST, term: t
```

MX-SPC3 Services Card

```
expo RT_FLOW: RT_FLOW_SESSION_POLICY_ACCEPT_USF: Tag SYSLOGMSG svc-set-name ss1:session created with policy accept 20.1.1.2/5->30.1.1.2/15100 0x0 icmp R11 1 sfw_policy1 ss1-ZoneIn ss1-ZoneOut 160000010 N/A(N/A) xe-5/3/2.0 UNKNOWN UNKNOWN UNKNOWN N/A N/A -1 N/A
```

Sample MX-SPC3 Output

Here's a sample output for MX-SPC3 card:

```
<14>1 2018-06-27T09:23:56.808-07:00 booklet RT_FLOW - RT_FLOW_SESSION_POLICY_ACCEPT_USF [junos@2636.1.1.1.2.25
prefix="PADDY-DEF" service-set-name="JNPR-NH-SSET3" source-address="50.0.0.10" source-port="1" destination-
address="60.0.0.10" destination-port="30170" connection-tag="0" service-name="icmp" rule-name="Tobe implemented"
rule-set-name="To be implemented" protocol-id="1" policy-name="p1" source-zone-name="JNPR-NH-SSET3-ZoneIn"
destination-zone-name="JNPR-NH-SSET3-ZoneOut" session-id-32="160000001" username="N/A" roles="N/A" packet-incoming-
interface="vms-2/0/0.100" application="UNKNOWN" nested-application="UNKNOWN" encrypted="UNKNOWN" application-
category="N/A" application-sub-category="N/A" application-risk="-1"] Prefix PADDY-DEF svc-set-name JNPR-NH-SSET3:
session created 50.0.0.10/1->60.0.0.10/30170 0x0 icmp To be implemented To be implemented 1 p1 JNPR-NH-SSET3-
ZoneIn JNPR-NH-SSET3-ZoneOut 160000001 N/A(N/A) vms-2/0/0.100 UNKNOWN UNKNOWN UNKNOWN N/A N/A -1
```

Stateful Firewall Rule Reject Logs

Following are example stateful firewall rule reject logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
Sep 20 01:42:02 mobst480b (FPC Slot 5, PIC Slot 0) 2017-09-20 08:41:31: SYSLOG_MSMP{SS_TEST}[jservices-sfw]:
JSERVICES_SFW_RULE_REJECT: proto 17 (UDP) application: any, 24.0.0.2:1234 -> 25.0.0.2:1234, Match SFW reject rule-
set: (null), rule: SFW_RULE_TEST, term: t
```

MX-SPC3 Services Card

```
expo RT_FLOW: RT_FLOW_SESSION_RULE_REJECT_USF: Tag SYSLOGMSG svc-set-name ss1: session denied 20.1.1.2/5-
>30.1.1.2/15183 0x0 icmp R11 1(8) sfw_policy1 ss1-ZoneIn ss1-ZoneOut UNKNOWN UNKNOWN N/A(N/A) xe-5/3/2.0 No
Rejected by policy 160000030 N/A N/A -1 N/A
```

Stateful Firewall Rule Discard Logs

Following are example stateful firewall rule discard logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
Sep 20 01:43:57 mobst480b (FPC Slot 5, PIC Slot 0) 2017-09-20 08:43:26: SYSLOG_MSMP{SS_TEST}[jservices-sfw]:
JSERVICES_SFW_RULE_DISCARD: proto 17 (UDP) application: any, 24.0.0.2:1234 -> 25.0.0.2:1234, Match SFW drop rule-
set: (null), rule: SFW_RULE_TEST, term: t
```

MX-SPC3 Services Card

```
RT_FLOW - RT_FLOW_SESSION_RULE_DISCARD_USF [junos@2636.1.1.1.2.25 tag="SYSLOG_SFW" service-set-name="ss1" source-
address="20.1.1.2" source-port="10000" destination-address="30.1.1.2" destination-port="20000" connection-tag="0"
service-name="None" rule-name="R1" rule-set-name="" protocol-id="17" icmp-type="0" policy-name="policy1" source-
zone-name="ss1-ZoneIn" destination-zone-name="ss1-ZoneOut" application="UNKNOWN" nested-application="UNKNOWN"
username="N/A" roles="N/A" packet-incoming-interface="xe-5/3/2.0" encrypted="No" reason="Denied by policy"
session-id-32="190000014" application-category="N/A" application-sub-category="N/A" application-risk="-1"
application-characteristics="N/A"] Tag SYSLOG_SFW svc-set-name ss1: session denied 20.1.1.2/10000->30.1.1.2/20000
0x0 None R1 17(0) policy1 ss1-ZoneIn ss1-ZoneOut UNKNOWN UNKNOWN N/A(N/A) xe-5/3/2.0 No Denied by policy 190000014
N/A N/A -1 N/A
```

Stateful Firewall Rule No Rule Drop Logs

Following are example stateful firewall rule no rule drop logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
Sep 20 01:43:57 mobst480b (FPC Slot 5, PIC Slot 0) 2017-09-20 08:43:26: SYSLOG_MSMP{SS_TEST}[jservices-sfw]:
JSERVICES_SFW_NO_RULE_DROP: proto 17 (UDP) application: any, 24.0.0.2:1234 -> 25.0.0.2:1234
```

MX-SPC3 Services Card

```
RT_FLOW_SESSION_NO_RULE_DROP_USF Prefix service-set-name protocol-id protocol-name source-interface-name separator
source-address source-port destination-address destination-port event-type;
```

Stateful Firewall No Policy Drop Logs

Following are example stateful firewall logs for MS-MPC services cards versus MX-SPC3 services processing card:

MS-MPC Services Card

```
JSERVICES_SFW_NO_POLICY source-address destination-address;
```

MX-SPC3 Services Card

```
RT_FLOW_SESSION_NO_POLICY_USF Prefix service-set-name source-address destination-address;
```

RELATED DOCUMENTATION

[Understanding Next Gen Services CGNAT Global System Logging | 110](#)

[Enabling Global System Logging for Next Gen Services | 112](#)

[Configuring Local System Logging for Next Gen Services | 113](#)

Configuring Syslog Events for NAT Rule Conditions with Next Gen Services

To configure syslog events to be generated when traffic matches NAT rule conditions for Next Gen Services NAT:

Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

The following are logs collected:

Out of addresses logs — If the allocation request fails to be handled as the public IP addresses in the No-PAT pool are used up, the out of addresses syslog is generated.

Out of ports logs — If the allocation request fails to be handled as the public IPs and ports in the NAPT pool are used up, the out of ports syslog is generated.

NAT Rule Match Logs — If the packet matches the NAT rule, the NAT rule match syslog is generated.

Pool resource release logs — If the public IP and port succeeds to be released to the NAPT pool, the pool release syslog is generated.

RELATED DOCUMENTATION

[Network Address Port Translation \(NAPT\) Overview | 171](#)

[Configuring Network Address Port Translation for Next Gen Services | 172](#)

Next Gen Services SNMP MIBS and Traps

IN THIS CHAPTER

- [Next Gen Services SNMP MIBs and Traps | 128](#)

Next Gen Services SNMP MIBs and Traps

IN THIS SECTION

- [Service-Set Related SNMP MIBs | 128](#)
- [Summary Mapping of MX-SPC3 CLI Services Operational Commands to SNMP MIBs | 136](#)
- [NAT SNMP MIBs | 141](#)
- [SNMP Traps | 144](#)

This topic describes the SNMP MIBS and traps for Next Gen Services with the MX-SPC3 services. As a reference, it also compares MX-SPC3 services card MIBS and traps with the MPC services card.

Service-Set Related SNMP MIBs

[Table 26 on page 129](#), [Table 27 on page 130](#), and [Table 28 on page 132](#) describe the MIB objects in the service-set related SNMP MIB tables supported in **jnxSPMIB**. This MIB is supported for both MS-MPC services cards and MX-SPC3 services cards with the exception of the following:

- The MX-SPC3 services card supports counters, such as memory usage and cpu usage, at the per service-set and per pic level, whereas MS-MPC services cards support these counters at the service level, for example, stateful firewall (SFW) and NAT).

The MX-SPC3 card uses the **jnxSpSvcSetTable** MIB for these counters.

- In [Table 26 on page 129](#) the **jnxSpSvcSetTable**, the object **jnxSpSvcSetSvcType** field will show a value of “ALL” since no per service-type specific counters are supported.

Table 26: Service-Set SNMP MIB Table (jnxSpSvcSetTable)

MIB Object	jnxSpSvcSet Entry Number	Description
jnxSpSvcSetIfName	jnxSpSvcSetEntry 4	The name of the interface identifying the AS PIC. If more than one interface is associated with the AS PIC, the name associated with the lower layer interface is used.
jnxSpSvcSetIfIndex	jnxSpSvcSetEntry 5	An index number associated with the interface name.
jnxSpSvcSetMemoryUsage	jnxSpSvcSetEntry 6	Amount of memory used by the service set, in bytes.
jnxSpSvcSetCpuUtil	jnxSpSvcSetEntry 7	Amount of CPU processing used by the service set, expressed as a percentage of total CPU usage.
jnxSpSvcSetSvcStyle	jnxSpSvcSetEntry 8	Type of service for the service set. Service types include: <ul style="list-style-type: none"> • Unknown—The service type is not known. • Interface-service—The service is interface based. • Next-hop-service—The service is next-hop based.
jnxSpSvcSetMemLimitPktDrops	jnxSpSvcSetEntry 9	Number of packets dropped because the service set exceeded its memory limits (operating in the Red zone).

Table 26: Service-Set SNMP MIB Table (jnxSpSvcSetTable) (Continued)

MIB Object	jnxSpSvcSet Entry Number	Description
jnxSpSvcSetCpuLimitPktDrops	jnxSpSvcSetEntry 10	Number of packets dropped because the service set exceeded the average CPU limits (when total CPU usage exceeds 85 percent).
jnxSpSvcSetFlowLimitPktDrops	jnxSpSvcSetEntry 11	Number of packets dropped because the service set exceeded the flow limit.
jnxSpSvcSetMemoryUsage64		Amount of memory used by the service set, in bytes.
jnxSpSvcSetMemLimitPktDrops64		Number of packets dropped because the service set exceeded its memory limits (operating in the Red zone).
jnxSpSvcSetCpuLimitPktDrops64		Number of packets dropped because the service set exceeded the average CPU limits (when total CPU usage exceeds 85 percent).
jnxSpSvcSetFlowLimitPktDrops64		Number of packets dropped because the service set exceeded the flow limit.
jnxSpSvcSetSessCount		Number of valid sessions in the service-set.

Table 27: Service-Set Service Type SNMP MIB Table (jnxSpSvcSetSvcTypeTable)

MIB Object	(jnxSpSvcSetSvcType Entry Number	Description
jnxSpSvcSetSvcTypeIndex	jnxSpSvcSetSvcTypeEntry 1	An integer used to identify the service type.

Table 27: Service-Set Service Type SNMP MIB Table (jnxSpSvcSetSvcTypeTable) (Continued)

MIB Object	(jnxSpSvcSetSvcType Entry Number	Description
jnxSpSvcSetSvcTypeIfName	jnxSpSvcSetSvcTypeEntry 2	The name of the interface identifying the AS PIC. If more than one interface is associated with the AS PIC, the name associated with the lower layer interface is used.
jnxSpSvcSetSvcTypeName	jnxSpSvcSetSvcTypeEntry 3	The name of the service type.
jnxSpSvcSetSvcTypeSvcSets	jnxSpSvcSetSvcTypeEntry 4	Number of service sets configured on the AS PIC that use this service type.
jnxSpSvcSetSvcTypeMemoryUsage	jnxSpSvcSetSvcTypeEntry 5	Amount of memory used by this service type, expressed in bytes.
jnxSpSvcSetSvcTypePctMemoryUsage	jnxSpSvcSetSvcTypeEntry 6	Amount of memory used by this service type, expressed as a percentage of total memory.
jnxSpSvcSetSvcTypeCpuUtil	jnxSpSvcSetSvcTypeEntry 7	Amount of CPU processing used by the service set, expressed as a percentage of total CPU usage.

Table 28: Service-Set Interface SNMP MIB Table (jnxSpSvcSetIfTable)

MIB Object	jnxSpSvcSetIf Entry Number	Description
jnxSpSvcSetIfTableName	jnxSpSvcSetIfEntry 1	The name of the interface used to identify the AS PIC. If more than one interface is associated with the AS PIC, the name associated with the lower layer interface is used.
jnxSpSvcSetIfSvcSets	jnxSpSvcSetIfEntry 2	The number of service sets configured on the AS PIC.
jnxSpSvcSetIfMemoryUsage	jnxSpSvcSetIfEntry 3	Amount of memory used by the AS PIC, expressed in bytes.
jnxSpSvcSetIfPctMemoryUsage	jnxSpSvcSetIfEntry 4	Amount of memory used by the AS PIC, expressed as a percentage of total memory.
jnxSpSvcSetIfPolMemoryUsage	jnxSpSvcSetIfEntry 5	Amount of policy memory used by the AS PIC, expressed in bytes.
jnxSpSvcSetIfPctPolMemoryUsage	jnxSpSvcSetIfEntry 6	Amount of policy memory used by the AS PIC, expressed as a percentage of the total.

Table 28: Service-Set Interface SNMP MIB Table (jnxSpSvcSetIfTable) (Continued)

MIB Object	jnxSpSvcSetIf Entry Number	Description
jnxSpSvcSetIfMemoryZone	jnxSpSvcSetIfEntry 7	<p>The memory usage zone currently occupied by the AS PIC. The definitions of each zone are:</p> <ul style="list-style-type: none"> • Green—All new flows are allowed. • Yellow—Unused memory is reclaimed. All new flows are allowed. • Orange—New flows are allowed only for service sets that use less than their equal share of memory. • Red—No new flows are allowed.
jnxSpSvcSetIfCpuUtil	jnxSpSvcSetIfEntry 8	Amount of CPU processing used by the AS PIC, expressed as a percentage of total CPU usage.
jnxSpSvcSetIfMemoryUsage64		Amount of policy memory used by the AS PIC, expressed in bytes.
jnxSpSvcSetIfPolMemoryUsage64		Amount of policy memory used by the AS PIC, expressed as a percentage of the total.
jnxSpSvcSetIfNumTotalSessActive		Total number of active sessions in the PIC.
jnxSpSvcSetIfPeakTotalSessActive		Number of active sessions in the PIC at any time.
jnxSpSvcSetIfNumCreatedSessPerSec		Number of created sessions per second in the PIC

Table 28: Service-Set Interface SNMP MIB Table (jnxSpSvcSetIfTable) (Continued)

MIB Object	jnxSpSvcSetIf Entry Number	Description
jnxSpSvcSetIfNumDeletedSessPerSec		Number of deleted sessions per second in the PIC
jnxSpSvcSetIfNumTotalTcpSessActive		Number of active sessions (TCP, UDP and other)in the PIC
jnxSpSvcSetIfNumTotalUdpSessActive		
jnxSpSvcSetIfNumTotalOtherSessActive		
jnxSpSvcSetIfPeakTotalTcpSessActive		Number of active sessions (TCP, UDP, and others) in the PIC
jnxSpSvcSetIfPeakTotalUdpSessActive		
jnxSpSvcSetIfPeakTotalOtherSessActive		
jnxSpSvcSetIfPeakCreatedSessPerSec		Number of created sessions per sec in the PIC
jnxSpSvcSetIfPeakDeletedSessPerSec		Number of deleted sessions per sec in the PIC
jnxSpSvcSetIfNumTotalTcplpv4SessActive		Total number of active sessions (TCP, UDP and other) for IPv4 and IPv6 in the PIC
jnxSpSvcSetIfNumTotalTcplpv6SessActive		
jnxSpSvcSetIfNumTotalUdplpv4SessActive		
jnxSpSvcSetIfNumTotalUdplpv6SessActive		
jnxSpSvcSetIfNumTotalOtherlpv4SessActive		
jnxSpSvcSetIfNumTotalOtherlpv6SessActive		

Table 28: Service-Set Interface SNMP MIB Table (jnxSpSvcSetIfTable) (Continued)

MIB Object	jnxSpSvcSetIf Entry Number	Description
jnxSpSvcSetIfNumTotalTcpGatedSessActive		Number of TCP and UDP gated sessions in the PIC
jnxSpSvcSetIfNumTotalUdpGatedSessActive		
jnxSpSvcSetIfNumTotalTcpRegSessActive		Number of TCP and UDP regular sessions in the PIC
jnxSpSvcSetIfNumTotalUdpRegSessActive		
jnxSpSvcSetIfNumTotalTcpTunSessActive		Number of TCP and UDP tunneled sessions in the PIC
jnxSpSvcSetIfNumTotalUdpTunSessActive		
jnxSpSvcSetIfSessPktRecv		Number of packets received in session handling
jnxSpSvcSetIfSessPktXmit		Number of packets transmitted as a part of session handling
jnxSpSvcSetIfSessSlowPathDiscard		Number of packets discarded in slow path
jnxSpSvcSetIfSessSlowPathForward		Number of packets forwarded in slow path
jnxSpSvcSetIfMspNumCreatedSubsPerSec		Number of subscribers created per sec
jnxSpSvcSetIfMspNumDeletedSubsPerSec		Number of Subscribers deleted per sec

Table 28: Service-Set Interface SNMP MIB Table (jnxSpSvcSetIfTable) (Continued)

MIB Object	jnxSpSvcSetIf Entry Number	Description
jnxSpSvcSetIfMspNumTotalSubsActive		Number of active subscribers
jnxSpSvcSetIfMspPeakCreatedSubsPer Sec		Peak number of created subscribers per sec in the PIC
jnxSpSvcSetIfMspPeakDeletedSubsPer Sec		Peak number of deleted subscribers per sec in the PIC
jnxSpSvcSetIfMspPeakTotalSubsActive		Peak number of total active subscribers in the PIC

Summary Mapping of MX-SPC3 CLI Services Operational Commands to SNMP MIBs

Table 29 on page 136 summarizes the mapping of the MX-SPC3 services card operations commands to the respective SNMP MIB.

Table 29: Summary Mapping of MX-SPC3 CLI Services Set Command to SNMP MIBs

CLI Command	Variable Name	MIB Tables	MIB Object
show services service-sets cpu-usage	cpu-utilization- percent	jnxSpSvcSetTable	jnxSpSvcSetCpuUtil
show services service-sets memory-usage	bytes-used		jnxSpSvcSetMemoryUsage64
show services service-sets memory-usage zone	mem-zone		jnxSpSvcSetIfMemoryZone
show services service-sets statistics packet-drops	cpulimit-drops		jnxSpSvcSetCpuLimitPktDrops 64

Table 29: Summary Mapping of MX-SPC3 CLI Services Set Command to SNMP MIBs (Continued)

CLI Command	Variable Name	MIB Tables	MIB Object
	flowlimit-drops		jnxSpSvcSetFlowLimitPktDrops64
	memlimit-drops		jnxSpSvcSetMemLimitPktDrops64
show services service-sets summary	service-set-bytes-used	jnxSpSvcSetIfTable	jnxSpSvcSetIfMemoryUsage64
	service-set-cpu-utilization		jnxSpSvcSetIfCpuUtil
	service-set-percent-bytes-used		jnxSpSvcSetIfPctMemoryUsage
	service-set-percent-policy-bytes-used		jnxSpSvcSetIfPctPolMemoryUsage
	service-set-policy-bytes-used		jnxSpSvcSetIfPolMemoryUsage64
	service-sets-configured		jnxSpSvcSetIfSvcSets
show services sessions count	sess-count	jnxSpSvcSetTable	jnxSpSvcSetSessCount
show services sessions analysis	num-total-session-active	jnxSpSvcSetIfTable	jnxSpSvcSetIfNumTotalSessActive
	peak-total-session-active		jnxSpSvcSetIfPeakTotalSessActive

Table 29: Summary Mapping of MX-SPC3 CLI Services Set Command to SNMP MIBs *(Continued)*

CLI Command	Variable Name	MIB Tables	MIB Object
	num-created-session-per-sec		jnxSpSvcSetIfNumCreatedSessPerSec
	num-deleted-session-per-sec		jnxSpSvcSetIfNumDeletedSessPerSec
	num-total-tcp-session-active		jnxSpSvcSetIfNumTotalTcpSessActive
	num-total-udp-session-active		jnxSpSvcSetIfNumTotalUdpSessActive
	peak-total-tcp-session-active		jnxSpSvcSetIfPeakTotalTcpSessActive
	peak-total-udp-session-active		jnxSpSvcSetIfPeakTotalUdpSessActive
	num-total-other-session-active		jnxSpSvcSetIfNumTotalOtherSessActive
	peak-created-session-per-second		jnxSpSvcSetIfPeakCreatedSessPerSec
	peak-deleted-session-per-second		jnxSpSvcSetIfPeakDeletedSessPerSec
	peak-total-other-session-active		jnxSpSvcSetIfPeakTotalOtherSessActive
	num-total-tcp-ipv4-session-active		jnxSpSvcSetIfNumTotalTcpIpv4SessActive

Table 29: Summary Mapping of MX-SPC3 CLI Services Set Command to SNMP MIBs (Continued)

CLI Command	Variable Name	MIB Tables	MIB Object
	num-total-tcp-ipv6-session-active		jnxSpSvcSetIfNumTotalTcplpv6SessActive
	num-total-udp-ipv4-session-active		jnxSpSvcSetIfNumTotalUdplpv4SessActive
	num-total-udp-ipv6-session-active		jnxSpSvcSetIfNumTotalUdplpv6SessActive
	num-total-tcp-gated-session-active		jnxSpSvcSetIfNumTotalTcpGatedSessActive
	num-total-udp-gated-session-active		jnxSpSvcSetIfNumTotalUdpGatedSessActive
	num-total-other-ipv4-session-active		jnxSpSvcSetIfNumTotalOtherIpv4SessActive
	num-total-other-ipv6-session-active		jnxSpSvcSetIfNumTotalOtherIpv6SessActive
	num-total-tcp-regular-session-active		jnxSpSvcSetIfNumTotalTcpRegSessActive
	num-total-udp-regular-session-active	jnxSpSvcSetIfTable	jnxSpSvcSetIfNumTotalUdpRegSessActive
	num-total-tcp-tunneled-session-active		jnxSpSvcSetIfNumTotalTcpTunSessActive

Table 29: Summary Mapping of MX-SPC3 CLI Services Set Command to SNMP MIBs (Continued)

CLI Command	Variable Name	MIB Tables	MIB Object
	num-total-udp-tunneled-session-active		jnxSpSvcSetIfNumTotalUdpTunSessActive
	session-pkts-received		jnxSpSvcSetIfSessPktRecv
	session-pkts-transmitted		jnxSpSvcSetIfSessPktXmit
	session-slow-path-discard		jnxSpSvcSetIfSessSlowPathDiscard
	session-slow-path-forward		jnxSpSvcSetIfSessSlowPathForward
show services subscriber analysis	msh-num-created-subs-per-sec		jnxSpSvcSetIfMshNumCreatedSubsPerSec
	msh-num-deleted-subs-per-sec		jnxSpSvcSetIfMshNumDeletedSubsPerSec
	msh-num-total-subs-active		jnxSpSvcSetIfMshNumTotalSubsActive
	msh-peak-created-subs-per-second		jnxSpSvcSetIfMshPeakCreatedSubsPerSec
	msh-peak-deleted-subs-per-second		jnxSpSvcSetIfMshPeakDeletedSubsPerSec
	msh-peak-total-subs-active		jnxSpSvcSetIfMshPeakTotalSubsActive

NAT SNMP MIBs

This section describes the **jnxSrcNatStatsTable** MIB objects.

[Table 30 on page 141](#) describes the source NAT SNMP MIB objects for the MS-MPC services card. This table exposes the source NAT translation attributes of the translated addresses.

[Table 31 on page 142](#) describes the source NAT SNMP MIB objects for the MX-SPC3 services card. This table contains information on source IP address translation only.

Table 30: MS-MPC Services Card Source NAT SNMP MIB Table (jnxSrcNatStatsTable)

jnxSrcNatStatsTable	MIB Object	Description
	jnxNatSrcPoolName	The name of dynamic source IP address pool
	jnxNatSrcXlatedAddrType	V4 or V6. The type of dynamic source IP address allocated from the address pool used in the NAT translation
	jnxNatSrcPoolType	The source port pool type indicates whether the address translation is done with port or without the port, or if it is a static translation. Ex napt-44, nat64 etc
	jnxNatSrcNumPortAvail	The number of ports available with this pool
	jnxNatSrcNumPortInuse	The number of ports in use for this NAT address entry
	jnxNatSrcNumAddressAvail	The total number of addresses available in this pool
	jnxNatSrcNumAddressInUse	The number of addresses in use from this pool
	jnxNatSrcNumSessions	The number of sessions are in use based on this NAT address entry

Table 30: MS-MPC Services Card Source NAT SNMP MIB Table (jnxSrcNatStatsTable) (Continued)

jnxSrcNatStatsTable	MIB Object	Description
jnxNatRuleTable		This table monitors NAT rule hits
	jnxNatRuleName	NAT rule name
	jnxNatRuleType	NAT types: Static Source, Static Destination, Dynamic Source and NAPT. Ex: napt44 etc
	jnxNatRuleTransHits	The number of hits on this NAT rule
jnxNatPoolTable		This table monitors NAT pool hits
	jnxNatPoolName	NAT Pool name
	jnxNatPoolType	NAT types: Static Source, Static Destination, Dynamic Source and NAPT. Ex: napt44 etc
	jnxNatPoolTransHits	The number of hits on this NAT Pool

Table 31: MX-SPC3 line cards, MX devices, and SRX devices Source NAT SNMP MIB Table (jnxNatObjects)

jnxJsSrcNatStatsTable	MIB Object	Description
	jnxJsNatSrcPoolName	The name of dynamic source IP address pool
	jnxJsNatSrcXlatedAddrType	New MIB. The type of dynamic source IP address allocated from the address pool used in the NAT translation. Value is v4 or v6

Table 31: MX-SPC3 line cards, MX devices, and SRX devices Source NAT SNMP MIB Table (jnxNatObjects) (Continued)

jnxJsSrcNatStatsTable	MIB Object	Description
	jnxJsNatSrcPoolType	withPAT or withoutPAT or static
	jnxJsNatSrcNumPortAvail	New MIB. The number of ports available with this pool
	jnxJsNatSrcNumPortInuse	The number of ports in use for this NAT address entry
	jnxJsNatSrcNumSessions	The number of sessions are in use based on this NAT address entry
	jnxJsNatSrcNumAddressAvail	New MIB. The total number of addresses available in this pool
	jnxJsNatSrcNumAddressInuse	New MIB. The number of addresses in use from this pool
jnxJsNatRuleTable		This table monitors NAT rule hits
	jnxJsNatRuleName	NAT rule name
	jnxJsNatRuleType	NAT types: Source, Destination and Static
	jnxJsNatRuleTransHits	The number of hits on this NAT rule. Status is deprecated. New - jnxJsNatRuleHits
	jnxJsNatRuleHits	The number of hits on this NAT rule,
	jnxJsNatRuleNumOfSessions	The number of sessions on this NAT rule
	jnxJsNatTransType	New MIB. Details below

Table 31: MX-SPC3 line cards, MX devices, and SRX devices Source NAT SNMP MIB Table (jnxNatObjects) (Continued)

jnxJsSrcNatStatsTable	MIB Object	Description
jnxJsNatPoolTable		This table monitors NAT pool hits
	jnxJsNatPoolName	NAT Pool name
	jnxJsNatPoolType	NAT types: Source, Destination and Static
	jnxJsNatPoolTransHits	The number of hits on this NAT Pool. (Only for SRX devices).
	jnxJsNatPoolUtility	Utilization of NAT Pool (Only for SRX devices).

SNMP Traps

Table 32 on page 144 describes the SNMP traps supported by both the MS-MPC services card and the MX-SPC3 services card.

Table 32: SNMP Traps

Trap	Description
SPD_TRAP_OIDS(jnxSpSvcSetZoneEntered)	jnxSpSvcSetZoneEntered – Indicates that an AS PIC has entered a more severe memory usage zone from a less severe memory usage zone. The zone entered is identified by JnxSpSvcSetIfMemoryZone
SPD_TRAP_OIDS(jnxSpSvcSetZoneExited)	jnxSpSvcSetZoneExited – Indicates that an AS PIC has exited a more severe memory usage zone to a less severe memory usage zone. The zone entered is identified by JnxSpSvcSetIfMemoryZone.

Table 32: SNMP Traps (Continued)

Trap	Description
SPD_TRAP_OIDS(jnxSpSvcSetCpuExceeded)	jnxSpSvcSetCpuExceeded — Indicates that an AS PIC has over 85% CPU usage.
SPD_TRAP_OIDS(jnxSpSvcSetCpuOk)	jnxSpSvcSetCpuOk — Indicates that an AS PIC has returned to less than 85%CPU usage.
SPD_TRAP_OIDS(jnxSpSvcSetFlowLimitUtilized)	jnxSpSvcSetFlowLimitUtilized — Indicates a service-set has reached its upper limit of flows threshold of a maximum flows allowed for a service set.

Configuring SNMP Trap Generation

This section describes how to configure the MS-MPC service card versus the MX-SPC3 services card to generate SNMP traps.

Configuring SNMP Trap for NAT Ports in a Source NAT Pool

If the current usage is above the raise threshold or below the clear threshold, we will generate a SNMP trap.

Configuring SNMP Traps for NAT Ports in a Source NAT Pool on an MS-MPC

```
user@host# set services nat pool NAT_POOL_TEST snmp-trap-thresholds address-port low 50
user@host# set services nat pool NAT_POOL_TEST snmp-trap-thresholds address-port high 75
```

Configuring SNMP Traps for NAT Ports in a Source NAT Pool on an MX-SPC3

```
user@host# set services nat source pool NAT_POOL_TEST pool-utilization-alarm raise-threshold 50
user@host# set services nat source pool NAT_POOL_TEST pool-utilization-alarm clear-threshold 40
```

Configuring SNMP Trap for Sessions

This is infra trap which configures SNMP flow thresholds for all flows for a service set or flows for all NAT pools configured for a service set.

Configuring a Sessions SNMP Trap on an MS-MPC

```
user@host# set services service-set SS_TEST max-flows 2m
user@host# set services service-set SS_TEST snmp-trap-thresholds flow low 50
user@host# set services service-set SS_TEST snmp-trap-thresholds flow high 75
```

Configuring a Sessions SNMP Trap on an MX-SPC3

```
user@host# set services service-set ss1 service-set-options session-limit maximum 2000
user@host# set services service-set ss1 snmp-trap-thresholds session low 50
user@host# set services service-set ss1 snmp-trap-thresholds session high 60
```

Example-Configuration for MX-SPC3 NAT for Three SNMP MIB Tables

Example Configuration

```
user@host> show services | display set
Configuration
=====
show services | display set

set services service-set ss1_nh_style1 nat-rule-sets rset1
set services service-set ss1_nh_style1 nat-rule-sets rset2
set services service-set ss1_nh_style1 nat-rule-sets rset5
set services service-set ss1_nh_style1 next-hop-service inside-service-interface vms-0/0/0.1
set services service-set ss1_nh_style1 next-hop-service outside-service-interface vms-0/0/0.2
set services nat source pool src_pool2_v6 address 300::0/128
set services nat source pool src_pool1 address 50.0.0.0/29
set services nat source rule-set rset1 rule nr1 match source-address 10.0.0.0/32
set services nat source rule-set rset1 rule nr1 match destination-address 20.0.0.0/32
set services nat source rule-set rset1 rule nr1 match application any
set services nat source rule-set rset1 rule nr1 then source-nat pool src_pool1
set services nat source rule-set rset1 match-direction input
set services nat source rule-set rset2 rule nr2_v6 match source-address 200::0/34
```

```

set services nat source rule-set rset2 rule nr2_v6 match destination-address 400::0/34
set services nat source rule-set rset2 rule nr2_v6 match application any
set services nat source rule-set rset2 rule nr2_v6 then source-nat pool src_pool2_v6
set services nat source rule-set rset2 match-direction input
set services nat destination pool src_pool5_dnat address 20.0.0.0/30
set services nat destination rule-set rset5 rule nr5_dnat match destination-address 21.0.0.0/30
set services nat destination rule-set rset5 rule nr5_dnat match application any
set services nat destination rule-set rset5 rule nr5_dnat then destination-nat pool
src_pool5_dnat
set services nat destination rule-set rset5 match-direction input
set services nat traceoptions file nat-trace.txt
set services nat traceoptions flag all

```

show snmp mib walk jnxJsSrcNatStatsTable

```

user@host>show snmp mib walk jnxJsSrcNatStatsTable
jnxJsNatSrcPoolName.2.112.49.0.0.0.0.0 = p1
jnxJsNatSrcXlatedAddrType.2.112.49.0.0.0.0.0 = 1
jnxJsNatSrcPoolType.2.112.49.0.0.0.0.0 = 1
jnxJsNatSrcNumPortInuse.2.112.49.0.0.0.0.0 = 0
jnxJsNatSrcNumSessions.2.112.49.0.0.0.0.0 = 0
jnxJsNatSrcNumPortAvail.2.112.49.0.0.0.0.0 = 10
jnxJsNatSrcNumAddressAvail.2.112.49.0.0.0.0.0 = 1
jnxJsNatSrcNumAddressInuse.2.112.49.0.0.0.0.0 = 0

```

show snmp mib walk jnxJsNatPoolTable

```

user@host>show snmp mib walk jnxJsNatPoolTable
jnxJsNatPoolName.9.115.114.99.95.112.111.111.108.49.1 = src_pool1
jnxJsNatPoolName.14.115.114.99.95.112.111.111.108.53.95.100.110.97.116.2 = src_pool5_dnat
jnxJsNatPoolType.9.115.114.99.95.112.111.111.108.49.1 = 1
jnxJsNatPoolType.14.115.114.99.95.112.111.111.108.53.95.100.110.97.116.2 = 2
jnxJsNatPoolTransHits.9.115.114.99.95.112.111.111.108.49.1 = 0
jnxJsNatPoolTransHits.14.115.114.99.95.112.111.111.108.53.95.100.110.97.116.2 = 0
jnxJsNatPoolHits.9.115.114.99.95.112.111.111.108.49.1 = 0
jnxJsNatPoolHits.14.115.114.99.95.112.111.111.108.53.95.100.110.97.116.2 = 0
jnxJsNatPoolUtil.9.115.114.99.95.112.111.111.108.49.1 = 0
jnxJsNatPoolUtil.14.115.114.99.95.112.111.111.108.53.95.100.110.97.116.2 = 0

```



```
Mar 20 15:07:52.575680 snmpd[0] <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
Mar 20 15:07:52.575697 snmpd[0] <<< V2 Trap
Mar 20 15:07:52.575714 snmpd[0] <<< Source:      10.48.12.170
Mar 20 15:07:52.575730 snmpd[0] <<< Destination: 190.1.1.1
Mar 20 15:07:52.575745 snmpd[0] <<< Version:     SNMPv2
Mar 20 15:07:52.575761 snmpd[0] <<< Community:   rtlogd_trap
Mar 20 15:07:52.575777 snmpd[0] <<<
Mar 20 15:07:52.575807 snmpd[0] <<< OID : sysUpTime.0
Mar 20 15:07:52.575824 snmpd[0] <<< type : TimeTicks
Mar 20 15:07:52.575841 snmpd[0] <<< value: (7605999) 21:07:39.99
Mar 20 15:07:52.575856 snmpd[0] <<<
Mar 20 15:07:52.575878 snmpd[0] <<< OID : snmpTrapOID.0
```

Memoryzone Trap

```
Mar 21 10:53:31.550471 snmpd[0] <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
Mar 21 10:53:31.550491 snmpd[0] <<< V2 Trap
Mar 21 10:53:31.550507 snmpd[0] <<< Source:      10.48.12.170
Mar 21 10:53:31.550522 snmpd[0] <<< Destination: 190.1.1.1
Mar 21 10:53:31.550536 snmpd[0] <<< Version:     SNMPv2
Mar 21 10:53:31.550551 snmpd[0] <<< Community:   rtlogd_trap
Mar 21 10:53:31.550566 snmpd[0] <<<
Mar 21 10:53:31.550585 snmpd[0] <<<    OID   : sysUpTime.0
Mar 21 10:53:31.550600 snmpd[0] <<<    type  : TimeTicks
Mar 21 10:53:31.550616 snmpd[0] <<<    value: (6076788) 16:52:47.88
Mar 21 10:53:31.550631 snmpd[0] <<<
Mar 21 10:53:31.550649 snmpd[0] <<<    OID   : snmpTrapOID.0
Mar 21 10:53:31.550664 snmpd[0] <<<    type  : Object
Mar 21 10:53:31.550681 snmpd[0] <<<    value: jnxSpSvcSetZoneEntered
Mar 21 10:53:31.550695 snmpd[0] <<<
Mar 21 10:53:31.550714 snmpd[0] <<<    OID   : jnxSpSvcSetIfMemoryZone.4294967295
Mar 21 10:53:31.550729 snmpd[0] <<<    type  : Number
Mar 21 10:53:31.550744 snmpd[0] <<<    value: 2
Mar 21 10:53:31.550758 snmpd[0] <<<
Mar 21 10:53:31.550776 snmpd[0] <<<    OID   : jnxSpSvcSetIfTableName.4294967295
Mar 21 10:53:31.550791 snmpd[0] <<<    type  : OctetString
```

[illegible]

Session Limit Trap

[illegible]

RELATED DOCUMENTATION

[Explore System Log Messages](#)

[SNMP MIB Explorer](#)

2

PART

Carrier Grade NAT (CGNAT)

- [Deterministic NAT Overview and Configuration | 154](#)
 - [Dynamic Address-Only Source NAT Overview and Configuration | 166](#)
 - [Network Address Port Translation Overview and Configuration | 171](#)
 - [NAT46 | 181](#)
 - [Stateful NAT64 Overview and Configuration | 185](#)
 - [IPv4 Connectivity Across IPv6-Only Network Using 464XLAT Overview and Configuration | 195](#)
 - [IPv6 NAT Protocol Translation \(NAT PT\) | 206](#)
 - [Stateless Source Network Prefix Translation for IPv6 Overview and Configuration | 209](#)
 - [Transitioning to IPv6 Using Softwires | 214](#)
 - [Transitioning to IPv6 Using DS-Lite Softwires | 220](#)
 - [Reducing Traffic and Bandwidth Requirements Using Port Control Protocol | 235](#)
 - [Transitioning to IPv6 Using MAP-E and MAP-T | 245](#)
 - [Monitoring and Troubleshooting Softwires | 265](#)
 - [Port Forwarding Overview and Configuration | 270](#)
 - [Port Translation Features Overview and Configuration | 279](#)
 - [Static Source NAT Overview and Configuration | 283](#)
 - [Static Destination NAT Overview and Configuration | 288](#)
 - [Twice NATPT Overview and Configuration | 293](#)
 - [Twice NAT Overview and Configuration | 303](#)
 - [Class of Service Overview and Configuration | 315](#)
-

Deterministic NAT Overview and Configuration

IN THIS CHAPTER

- [Deterministic NAT Overview for Next Gen Services | 154](#)
- [Configuring Deterministic NAT for Next Gen Services | 160](#)

Deterministic NAT Overview for Next Gen Services

IN THIS SECTION

- [Benefits of Deterministic NAT | 155](#)
- [Understanding Deterministic NAT Algorithms | 155](#)
- [Deterministic NAT Restrictions | 159](#)

Under Next Gen Services with the MX-SPC3, you can configure both Deterministic NAT44 and NAT64 services. Next Gen Services deterministic NAT services use an algorithm to allocate blocks of destination ports.

Next Gen Services deterministic NAT44 service ensures that the original source IPv4 address and port always map to the same post-NAT IPv4 address and port range, and that the reverse mapping of a given translated external IPv4 address and port are always mapped to the same internal IPv4 address.

Next Gen Services deterministic NAT64 service ensures that the original source IPv6 address and port always map to the same post-NAT IPv4 address and port range, and that the reverse mapping of a given translated external IPv4 address and port are always mapped to the same internal IPv6 address.

For detailed information on how to configure deterministic NAT, see "[Configuring Deterministic NAT for Next Gen Services](#)" on page 160.

Benefits of Deterministic NAPT

- Eliminates the need for address translation logging because an IP address is always mapped to the same external IP address and port range, and the reverse mapping of a given translated external IP address and port are always mapped to the same internal IP address.

Understanding Deterministic NAPT Algorithms

The effectiveness of your implementation of deterministic NAPT depends on your analysis of your subscriber requirements. The block size you provide indicates how many ports will be made available for each incoming subscriber address from the range in the `from` clause specified in the applicable NAT rule. The allocation algorithm computes an offset value to determine the outgoing IP address and port. A reverse algorithm is used to derive the originating subscriber address.



NOTE: In order to track subscribers without using logs, an ISP must use a reverse algorithm to derive a subscriber (source) addresses from a translated address.

The following variables are used in forward calculation (private subscriber IP address to public IP address) and reverse calculation (public IP address to private subscriber IP address):

- `Pr_Prefix`—Any pre-NAT IPv4 subscriber address.
- `Pr_Port`—Any pre-NAT protocol port.
- `Block_Size`—Number of ports configured to be available for each `Pr_Prefix`.

If `block-size` is configured as zero, the method for computing the block size is computed as follows:

$$\text{block-size} = \text{int}(64512 / \text{ceil}[(\text{Nr_Addr_PR_Prefix} / \text{Nr_Addr_PU_Prefix})])$$

where 64512 is the maximum available port range per public IP address.

- `Base_PR_Prefix`—First usable pre-NAT IPv4 subscriber address in a `from` clause of the NAT rule.
- `Base_PU_Prefix`—First usable post-NAT IPv4 subscriber address configured in the NAT pool.
- `Pu_Port_Range_Start`—First usable post-NAT port. This is 1024.
- `Pr_Offset`—The offset of the pre-NAT IP address that is being translated from the first usable pre-NAT IPv4 subscriber address in a `from` clause of the NAT rule. $\text{Pr_Offset} = \text{Pr_Prefix} - \text{Base_Pr_Prefix}$.
- `PR_Port_Offset`—Offset of the pre-NAT IP address multiplied by the block size. $\text{PR_Port_Offset} = \text{Pr_Offset} * \text{Block_Size}$.
- `Pu_Prefix`—Post-NAT address for a given `Pr_Prefix`.

- Pu_Start_Port—Post-NAT start port for a flow from a given Pr_Prefix
- Pu_Actual_Port—Post-NAT port seen on a reverse flow.
- Nr_Addr_PR_Prefix — Number of usable pre-NAT IPv4 subscriber addresses in a from clause of the NAT rule.
- Nr_Addr_PU_Prefix — Number of usable post-NAT IPv4 addresses configured in the NAT pool.
- Rounded_Port_Range_Per_IP — Number of ports available for each post-NAT IP address.

$$\text{Rounded_Port_Range_Per_IP} = \text{ceil}[(\text{Nr_Addr_PR_Prefix}/\text{Nr_Addr_PU_Prefix})] * \text{Block_Size}.$$
- Pu_Offset—Offset of the post-NAT IP address from the first usable post-NAT address. $\text{Pu_Offset} = \text{Pu_Prefix} - \text{Base_Pu_Prefix}.$
- Pu_Port_Offset— Offset of the post-NAT port from 1024 added to the product of the offset of the post-NAT IP address and the number of ports available for each post-NAT IP address.

$$\text{Pu_Port_Offset} = (\text{Pu_Offset} * \text{Rounded_Port_Range_Per_IP}) + (\text{Pu_Actual_Port} - \text{Pu_Port_Range_Start}).$$

Algorithm Usage—Assume the following configurations:

```

services {
  nat {
    source {
      pool src-pool {
        address 203.0.113.0/16;
        port {
          automatic {
            random-allocation;
          }
          deterministic {
            block-size 249;
            host address 10.1.0.1/16;
          }
        }
      }
    }
  }
  rule-set set1 {
    rule det-nat {
      match-direction input;
      match {
        source-address 10.1.0.0/16;
      }
      then {

```

```

source-nat {
    pool src-pool;
}
}
}
}
}
}
}
}
}
}

```

Forward Translation

1. $Pr_Offset = Pr_Prefix - Base_Pr_Prefix - \text{gaps in the Private IPs pool}$



NOTE: When the Private IPs pool is made of several pools that are not contiguous, the Pr_Offset must count only the Private IPs in the pools. So it is the sum of:

- The offset within the pool where the IP falls into.
- The size of the pools with lower IPs.

2. $Pr_Port_Offset = Pr_Offset * Block_Size$
3. $Rounded_Port_Range_Per_IP = \text{ceil}[(Nr_Addr_PR_Prefix / Nr_Addr_PU_Prefix)] * Block_Size$
4. $Pu_Prefix = Base_Public_Prefix + \text{floor}(Pr_Port_Offset / Rounded_Port_Range_Per_IP)$



NOTE: When the Public IPs pool is made of several pools that are not contiguous, the Pu_Offset must count only the Public IPs in the pools. So the sum must be intended as:

- If the value $\text{floor}(Pr_Port_Offset / Rounded_Port_Range_Per_IP)$ is greater than the size of the first Public IP pool, subtract the size of this first pool from the value. Then, consider the second pool size.
- Repeat the process until the value is lesser than the n-th pool.

5. $Pu_Start_Port = Pu_Port_Range_Start + (Pr_Port_Offset \% Rounded_Port_Range_Per_IP)$

Using the sample configuration and assuming a subscriber flow sourced from 10.1.1.250:5000:

1. $Pr_Offset = 10.1.1.250 - 10.1.0.1 = 505$
2. $Pr_Port_Offset = 505 * 249 = 125,745$

$$3. \text{Rounded_Port_Range_Per_IP} = \text{ceil}[(65,533/254)] * 249 = 259 * 249 = 64,491$$

$$4. \text{Pu_Prefix} = 203.0.113.1 + \text{floor}(125,745 / 64,491) = 203.0.113.1 + 1 = 203.0.113.2$$

$$5. \text{Pu_Start_Port} = 1,024 + (125,745 \% 64,491) = 62278$$

- 10.1.1.250 is translated to 203.0.113.2.
- The starting port is 62278. There are 249 ports available to the subscriber based on the configured block size. The available port range spans ports 62278 through 62526 (inclusive).
- The specific flow 10.1.1.250:5000 randomly assigns any of the ports in its range because random allocation was specified.

Reverse Translation

$$1. \text{Pr_Offset} = \text{Pr_Prefix} - \text{Base_Pr_Prefix} - \text{gaps in the Private IPs pool}$$



NOTE: When the Private IPs pool is made of several pools that are not contiguous, the Pr_Offset must count only the Private IPs in the pools. So it is the sum of:

- The offset within the pool where the IP falls into.
- The size of the pools with lower IPs.

$$2. \text{Pu_Port_Offset} = (\text{Pu_Offset} * \text{Rounded_Port_Range_Per_IP}) + (\text{Pu_Actual_Port} - \text{Pu_Port_Range_Start})$$

$$3. \text{Subscriber_IP} = \text{Base_Pr_Prefix} + \text{floor}(\text{Pu_Port_Offset} / \text{Block_Size})$$

The reverse translation is determined as follows. Assume a flow returning to 203.0.113.2:62278.

$$1. \text{Pu_Offset} = 203.0.113.2 - 203.0.113.1 = 1$$

$$2. \text{Pu_Port_Offset} = (1 * 64,491) + (62,280 - 1024) = 125,747$$

$$3. \text{Subscriber_IP} = 10.1.0.1 + \text{floor}(125,747 / 249) = 10.1.0.1 + 505 = 10.1.1.250$$



NOTE: In reverse translation, only the original private IP address can be derived, and not the original port in use. This is sufficiently granular for law enforcement requirements.

When you have configured deterministic NAT, you can use the `show services nat deterministic-nat internal-host` and `show services nat deterministic-nat nat-port-block` commands to show forward and reverse mapping. However, mappings will change if you reconfigure your deterministic port block allocation


block size or the `from` clause for your NAT rule. In order to provide historical information on mappings, we recommend that you write scripts that can show specific mappings for prior configurations.

Deterministic NAT Restrictions

When you configure deterministic NAT, be aware of the following:

- For IPv6 deterministic NAT64 host address configuration, we support the last 32-bit (4 byte) change of the IPv6 host prefix. This means we only can configure /96 prefix masks for IPv6 address, which supports a maximum address number of 2^{32} for one IPv6 prefix. The host address is specified at the `[services nat source pool p1 port deterministic host]` configuration hierarchy.

- Usually, the number of address in host-range should be more than the number of address in pool.


-  **BEST PRACTICE:** We don't recommend the host address number be configured to exceed the total port block resource number because some hosts may not receive a port block resource successfully.

- The minimum block size for deterministic NAT is 1. If you configure a smaller block size, the commit fails. If the block size is configured to 0, the block size will be automatically calculated based on host number and translated address number. If the calculated block size is less than 1, the commit fails.

- For Next Gen Services deterministic NAT, you can configure a mix of IPv4 and IPv6 host addresses together in a NAT pool in either a host address or an address name list, However. the total host prefix number cannot exceed 1000.

- You cannot configure an address range or DNS name in a host address book name.

- The configured host address prefix and host address book name are merged together if its prefixes are overlapped. You can use the `show services nat source deterministic operational` command to show the merged prefixes.

-  **BEST PRACTICE:** We recommend, you keep subscriber host addresses consistent with multiple rule's matching the source address prefix, if the same deterministic NAT pool is used across multiple rules; otherwise, traffic from hosts which are not configured in the NAT pool, even it matches the NAT rule, may not allocate the port successfully.

- For Next Gen Services NAT services, the total number of host addresses configured must be greater than or equal to the deterministic NAT port blocks available.

RELATED DOCUMENTATION

[Configuring Deterministic NAT for Next Gen Services | 160](#)

Configuring Deterministic NAT for Next Gen Services

IN THIS SECTION

- [Configuring the NAT Pool for Deterministic NAT for Next Gen Services | 160](#)
- [Configuring the NAT Rule for Deterministic NAT44 for Next Gen Services | 162](#)
- [Configuring the NAT Rule for Deterministic NAT64 for Next Gen Services | 163](#)
- [Configuring the Service Set for Deterministic NAT for Next Gen Services | 164](#)
- [Clearing the Don't Fragment Bit | 165](#)

Deterministic NAT for Next Gen Services is available only for MX series devices. To configure deterministic NAT on Next Gen Services, perform the following:

Configuring the NAT Pool for Deterministic NAT for Next Gen Services

To configure the NAT pool for deterministic NAT:

1. Create a pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix to address address-prefix
```

3. Configure deterministic port block allocation for the pool.

```
[edit services nat source pool nat-pool-name port]
user@host# set deterministic
```

4. If you want the lowest and highest IPv4 addresses (the network and broadcast addresses) in the source address range of a NAT rule to be translated when the NAT pool is used, configure `include-boundary-address`.

```
[edit services nat source pool nat-pool-name port deterministic]
user@host# set include-boundary-addresses
```

5. Configure the port block size. The range is 1 to 64,512. The default block size is 256.

```
[edit services nat source pool nat-pool-name port deterministic]
user@host# set block-size block-size
```

6. Configure the first usable pre-NAT subscriber address, which is used in calculating the offset value for a pre-NAT address that is being translated. This offset is used to perform the deterministic NAT mapping.

```
[edit services nat source pool nat-pool-name port deterministic]
user@host# set host address host-addr
```

7. Configure the interval at which the syslog is generated for the deterministic NAT configuration.

```
[edit services nat source pool nat-pool-name port deterministic]
user@host# set deterministic-nat-configuration-log-interval seconds
```

8. To configure automatic port assignment for the pool, specify either random allocation or round-robin allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set automatic (random-allocation | round-robin)
```

Random allocation randomly assigns a port from the range 1024 through 65535 for each port translation. Round robin allocation first assigns port 1024, and uses the next higher port for each successive port assignment. Round robin allocation is the default.

9. To disable round-robin port allocation for all NAT pools that do not specify an automatic (random-allocation | round-robin) setting, configure the global setting.

```
[edit services nat source]
user@host# set port-round-robin disable
```

SEE ALSO

| *Network Address Translation Configuration Overview*

Configuring the NAT Rule for Deterministic NAPT44 for Next Gen Services

To configure the NAT rule for deterministic NAPT44:

1. Configure the NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
```

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify one or more application protocols to which the NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Specify the NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

Configuring the NAT Rule for Deterministic NAPT64 for Next Gen Services

To configure the NAT rule for deterministic NAPT64:

1. Configure the source NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the IPv6 prefix for the source addresses that are translated by the NAT rule.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```


4. Specify one or more application protocols to which the NAT rule applies. The number of application terms must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Specify the NAT source pool that contains the addresses for translated source addresses.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

Configuring the Service Set for Deterministic NAT for Next Gen Services

To configure the service set for deterministic NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Clearing the Don't Fragment Bit

If you configured deterministic NAPT64, specify that the don't fragment (DF) bit for IPv4 packet headers is cleared when the packet length is less than 1280 bytes.

```
[edit services nat natv6v4]  
user@host# set clear-dont-fragment-bit
```

This prevents unnecessary creation of an IPv6 fragmentation header when translating IPv4 packets that are less than 1280 bytes.

RELATED DOCUMENTATION

| [Deterministic NAPT Overview for Next Gen Services](#) | 154

Dynamic Address-Only Source NAT Overview and Configuration

IN THIS CHAPTER

- [Dynamic Address-Only Source Translation Overview | 166](#)
- [Configuring Dynamic Address-Only Source NAT for Next Gen Services | 167](#)

Dynamic Address-Only Source Translation Overview

IN THIS SECTION

- [Benefits of Dynamic Address-Only Source Translation | 166](#)

With dynamic address-only translation, you can map a private IP source address to a public IP address. A public address is picked up dynamically from a source NAT pool, and the mapping from the original source address to the translated source address is maintained as long as there is at least one active flow that uses this mapping. The port is not mapped.

Benefits of Dynamic Address-Only Source Translation

- Allows hosts in the private network to connect with the external domain, while hiding the private network.
- Allows a few public IP addresses to be used by several private hosts

RELATED DOCUMENTATION

| [Configuring Dynamic Address-Only Source NAT for Next Gen Services | 167](#)

Configuring Dynamic Address-Only Source NAT for Next Gen Services

IN THIS SECTION

- [Configuring the Source Pool for Dynamic Address-Only Source NAT | 167](#)
- [Configuring the NAT Source Rule for Dynamic Address-Only Source NAT | 168](#)
- [Configuring the Service Set for Dynamic Address-Only Source NAT | 170](#)

Configuring the Source Pool for Dynamic Address-Only Source NAT

To configure the source pool for dynamic address-only source NAT:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix to address address-prefix
```



NOTE: The first and last address of the IP pool must be configured with /32 prefix.

3. Disable port translation.

```
[edit services nat source pool nat-pool-name]  
user@host# set port no-translation
```

4. Define the NAT pool utilization levels that trigger SNMP traps. The raise-threshold is the pool utilization percentage that triggers the trap, and the range is 50 through 100. The clear-threshold is the pool utilization percentage that clears the trap, and the range is 40 through 100. The utilization is based on the number of addresses that are used.

```
[edit services nat source pool nat-pool-name]
user@host# set pool-utilization-alarm raise-threshold value
user@host# set pool-utilization-alarm clear-threshold value
```

If you do not configure pool-utilization-alarm, traps are not created.

5. To allow the IP addresses of a NAT source pool or destination pool to overlap with IP addresses in pools used in other service sets, configure allow-overlapping-pools.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Source Rule for Dynamic Address-Only Source NAT

To configure the NAT source rule for dynamic address-only source NAT:

1. Configure the NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify one or more application protocols to which the NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Specify the NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

6. Configure the address-pooling paired feature if you want to ensure assignment of the same external IP address for all sessions originating from the same internal host.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat mapping-type]
user@host# set address-pooling-paired
```

7. Specify the timeout period for address-pooling-paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

8. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Dynamic Address-Only Source NAT

To configure the service set for dynamic address-only source NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

RELATED DOCUMENTATION

[Dynamic Address-Only Source Translation Overview](#) | 166

Network Address Port Translation Overview and Configuration

IN THIS CHAPTER

- [Network Address Port Translation \(NAPT\) Overview | 171](#)
- [Configuring Network Address Port Translation for Next Gen Services | 172](#)
- [Configuring Syslog Events for NAT Rule Conditions with Next Gen Services | 179](#)

Network Address Port Translation (NAPT) Overview

IN THIS SECTION

- [Benefits of NAPT | 172](#)

NAPT translates a private source IP address to an external source address and port. Multiple private IP addresses can be mapped to the same external address because each private address is mapped to a different port of the external address.

With NAPT, you can configure up to 32 external address ranges, and map up to 65,536 private addresses to each external address.

NAPT supports the following:

- Round-robin port and address allocation (see ["Round-Robin Port Allocation" on page 281](#)).
- Address pooling and endpoint independent mapping (see ["Address Pooling and Endpoint Independent Mapping for Port Translation" on page 279](#)).
- Secured port block allocation (see ["Secured Port Block Allocation for Port Translation" on page 282](#)).

Benefits of NAPT

- Allows hosts in the private network to connect with the external domain, while hiding the private network.
- Minimizes the number of public IP addresses that are allocated for NAT.

Configuring Network Address Port Translation for Next Gen Services

IN THIS SECTION

- [Configuring the Source Pool for NAPT | 172](#)
- [Configuring the NAT Source Rule for NAPT | 176](#)
- [Configuring the Service Set for NAPT | 178](#)

Configuring the Source Pool for NAPT

To configure the source pool for NAPT:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix to address address-prefix
```

3. To configure automatic port assignment for the pool, specify either random allocation or round-robin allocation. Round-robin allocation is the default.

```
[edit services nat source pool nat-pool-name port]
user@host# set automatic (random-allocation | round-robin)
```

Random allocation randomly assigns a port from the range 1024 through 65535 for each port translation. Round-robin allocation first assigns port 1024, and uses the next higher port for each successive port assignment.

4. To disable round-robin port allocation for all NAT pools that do not specify an automatic (random-allocation | round-robin) setting, configure the global setting.

```
[edit services nat source]
user@host# set port-round-robin disable
```

5. To configure a range of ports to assign to a pool, perform the following:



NOTE: If you specify a range of ports to assign, the automatic statement is ignored.

- a. Specify the low and high values for the port. If you do not configure automatic port assignment, you must configure a range of ports.

```
[edit services nat source pool nat-pool-name port]
user@host# set range port-low to port-high
```

- b. Specify either random allocation or round-robin allocation. Round-robin allocation is the default.

```
[edit services nat source pool nat-pool-name port range]
user@host# set (random-allocation | round-robin)
```

6. Assign a port within the same range as the incoming port—either 0 through 1023 or 1024 through 65,535. This feature is not available if you configure port-block allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set preserve-range
```

7. Assign a port with the same parity (even or odd) as the incoming source port. This feature is not available if you configure port-block allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set preserve-parity
```

8. Configure a global default port range for NAT pools that use port translation. This port range is used when a NAT pool does not specify a port range and does not specify automatic port assignment. The global port range can be from 1024 through 65,535.

```
[edit services nat source]
user@host# set pool-default-port-range port-low to port-high
```

9. If you want to allocate a block of ports for each subscriber to use for NAPT, configure port-block allocation:
 - a. Configure the number of ports in a block. The range is 1 through 64,512 and the default is 128.

```
[edit services nat source pool nat-pool-name port]
user@host# set block-allocation block-size block-size
```

- b. Configure the interval, in seconds, for which the block is active. After the timeout, a new block is allocated, even if ports are available in the active block. If you set the timeout to 0, port blocks are filled completely before a new port block is allocated, and the last port block remains active indefinitely. The range is 0 through 86,400, and the default is 0.

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set active-block-timeout timeout-interval
```

- c. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

- d. Configure the maximum number of blocks that can be allocated to a user address. The range is 1 through 512, and the default is 8.

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set maximum-blocks-per-host maximum-block-number
```

- e. Specify how often to send interim system logs for active port blocks and for inactive port blocks with live sessions. This increases the reliability of system logs, which are UDP-based and can get lost in the network. The range is 1800 through 86,400 seconds, and the default is 0 (interim logs are disabled).

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set interim-logging-interval timeout-interval
```

10. Specify the timeout period for endpoint independent translations that use the specified NAT pool. Mappings that are inactive for this amount of time are dropped. The range is 120 through 86,400 seconds. If you do not configure `ei-mapping-timeout`, then the `mapping-timeout` value is used for endpoint independent translations.

```
[edit services nat source pool nat-pool-name]
user@host# set ei-mapping-timeout ei-mapping-timeout
```

11. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

12. Define the NAT pool utilization levels that trigger SNMP traps. The `raise-threshold` is the pool utilization percentage that triggers the trap, and the range is 50 through 100. The `clear-threshold` is the pool utilization percentage that clears the trap, and the range is 40 through 100. For pools that use port-block allocation, the utilization is based on the number of ports that are used; for pools

that do not use port-block allocation, the utilization is based on the number of addresses that are used.

```
[edit services nat source pool nat-pool-name]
user@host# set pool-utilization-alarm raise-threshold value
user@host# set pool-utilization-alarm clear-threshold value
```

If you do not configure `pool-utilization-alarm`, traps are not created.

13. To allow the IP addresses of a NAT pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`. However, pools that configure port-block allocation must not overlap with other pools.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Source Rule for NAPT

To configure the NAT source rule for NAPT:

1. Configure the NAT rule name.

```
[edit services nat source]
user@host# edit rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the source addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat source rule-set rule-set-name rule rule-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify one or more application protocols to which the NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Specify the NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

6. Configure the address-pooling paired feature if you want to ensure assignment of the same external IP address for all sessions originating from the same internal host.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat mapping-type]
user@host# set address-pooling
```

7. If you want to ensure that the same external address and port are assigned to all connections from a given host, configure endpoint-independent mapping:
 - a. Configure the mapping type as endpoint independent.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set mapping-type endpoint-independent
```

- b. Specify prefix lists that contain the hosts that are allowed to establish inbound connections using the endpoint-independent mapping. (Prefix lists are configured at the [edit policy-options] hierarchy level.)

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set filtering-type endpoint-independent prefix-list [allowed-host] except
[denied-host]
```

- c. Specify the maximum number of inbound flows allowed simultaneously on an endpoint-independent mapping.

```
[edit services nat source rule-set rule-set-name rule rule-name filtering-type then source-
nat]
user@host# set secure-nat-mapping eif-flow-limit number-of-flows
```

- d. Specify the direction in which active endpoint-independent mapping is refreshed. By default, mapping is refreshed for both inbound and outbound active flows.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set secure-nat-mapping mapping-refresh (inbound | inbound-outbound | outbound)
```

8. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for NAT

To configure the service set for NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

RELATED DOCUMENTATION

[Network Address Port Translation \(NAPT\) Overview](#) | 171

Configuring Syslog Events for NAT Rule Conditions with Next Gen Services

To configure syslog events to be generated when traffic matches NAT rule conditions for Next Gen Services NAT:

Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

The following are logs collected:

Out of addresses logs — If the allocation request fails to be handled as the public IP addresses in the No-PAT pool are used up, the out of addresses syslog is generated.

Out of ports logs – If the allocation request fails to be handled as the public IPs and ports in the NAT pool are used up, the out of ports syslog is generated.

NAT Rule Match Logs – If the packet matches the NAT rule, the NAT rule match syslog is generated.

Pool resource release logs – If the public IP and port succeeds to be released to the NAT pool, the pool release syslog is generated.

RELATED DOCUMENTATION

[Network Address Port Translation \(NAPT\) Overview | 171](#)

[Configuring Network Address Port Translation for Next Gen Services | 172](#)

NAT46

IN THIS CHAPTER

- [NAT46 Next Gen Services Configuration Examples | 181](#)

NAT46 Next Gen Services Configuration Examples

IN THIS SECTION

- [NAT46 Support Summary | 182](#)
- [NAT46 Sample Configuration | 183](#)

Starting in Junos OS Release 20.2R1 you can run NAT46 Next Gen Services.

Starting in Junos OS Release 20.2R1, Network Address Translation and Protocol Translation (NAT-PT) [RFC2766] are supported for CGNAT Next Gen Services. NAT46 is a IPv4-to-IPv6 transition mechanism that provides a way for end-nodes in IPv6 realm to communicate with end-nodes in IPv4 realm and vice versa. This is achieved using a combination of Network Address Translation and Protocol Translation.

NAT46 is supported on both the SRX and on MX240, MX480, and MX960 for CGNAT Next Gen Services. This topic provides example configurations to help you understand how to configure NAT46 CGNAT Next Gen Services on these MX Series routers.



NOTE: These examples are for SRX Series Firewalls. However, you can use these same examples to configure NAT46 Next Gen Services on MX Series devices. Use the configuration statements under the [edit services....] hierarchy on MX Series devices to configure NAT46 Next Gen Services.

You can find these examples here: [IPv6 NAT](#)

There are four examples available:

- Configuring an IPv4-Initiated Connection to an IPv6 Node Using Default Destination Address Prefix Static Mapping — This example shows how to configure an IPv4-initiated connection to an IPv6 node using default destination address prefix static mapping.
- Configuring an IPv4-Initiated Connection to an IPv6 Node Using Static Destination Address One-to-One Mapping — This example shows how to configure an IPv4-initiated connection to an IPv6 node using static destination address one-to-one mapping.
- Configuring an IPv6-Initiated Connection to an IPv4 Node Using Default Destination Address Prefix Static Mapping — This example shows how to configure an IPv6-initiated connection to an IPv4 node using default destination address prefix static mapping. This example does not show how to configure the NAT translation for the reverse direction.
- Configuring an IPv6-Initiated Connection to an IPv4 Node Using Static Destination Address One-to-One Mapping — This example shows how to configure an IPv6-initiated connection to an IPv4 node using static destination address one-to-one mapping.

NAT46 Support Summary

NAT46 for Next Gen Services supports the following:

- ICMP, TCP, and UDP protocol packets.
- Static mapping is used to communicate between the IPv4 to IPv6 side of the subscriber connection.
- Bi-directional traffic flow is supported if you have other ways to convey the mapping between the IPv6 address and the dynamically allocated IPv4 address.
- NAT46 supports DNS, ICMP, and FTP ALGs.

Keep these things in mind when configuring NAT46 for Next Gen Services:

- No support of NAT64 feature described in NAT-PT (RFC 2765).
- Static NAT is not used for the source translation in any NAT scenario.
- Except DNS, FTP and ICMP, other ALGs are not supported for NAT46.
- AMS functionality is not supported for NAT46.
- Port translation is not tested with Source Address NAT (when source pool is a IPv6 prefix) for the NAT46 feature.

NAT46 Sample Configuration

This sample configuration applies for MX Series devices:

```

services {
  nat {
    source {
      pool ipv6_prefix {
        address 27a6::/96;
      }
    }
    rule-set myipv6_rs {
      rule ipv6_rule {
        match {
          source-address 10.1.1.1/30 ;
          destination-address 27a6::a0a:a2d/126;
        }
        then {
          source-nat {
            pool {
              ipv6_prefix;
            }
          }
        }
      }
    }
    match-direction input;
  }
}

static {
  rule-set test_rs {
    rule test_rule {
      match {
        destination-address ip-address;
      }
      then {
        static-nat {
          prefix ip-address;
        }
      }
    }
  }
}

.....match-direction input;
}

```

```
    }
    service-set sset1 {
        ...
        nat-rule-sets test_rs;
        nat-rule-sets myipv6_rs;
        ...
    }
}
```

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.2R1	Starting in Junos OS Release 20.2R1 you can run NAT46 Next Gen Services.
20.2R1	Starting in Junos OS Release 20.2R1, Network Address Translation and Protocol Translation (NAT-PT) [RFC2766] are supported for CGNAT Next Gen Services.

RELATED DOCUMENTATION

<i>service-set</i>
Configuring Service Sets for Network Address Translation

Stateful NAT64 Overview and Configuration

IN THIS CHAPTER

- [Stateful NAT64 Overview | 185](#)
- [IPv4 Addresses Embedded in IPv6 Addresses | 186](#)
- [Configuring Next Gen Services Stateful NAT64 | 187](#)

Stateful NAT64 Overview

IN THIS SECTION

- [Benefits of Stateful NAT64 | 185](#)

Stateful NAT64 translates IPv6 addresses to public IPv4 addresses, allowing IPv6-only clients to contact IPv4 servers using unicast UDP, TCP, or ICMP. Stateful NAT64 translates the destination IPv6 address to the embedded IPv4 address, and translates the source IPv6 address to a public IPv4 address and port from a block of IPv4 addresses that you set aside.

Stateful NAT64 supports the following:

- Round-robin port and address allocation (see ["Round-Robin Port Allocation" on page 281](#)).
- Address pooling and endpoint independent mapping (see ["Address Pooling and Endpoint Independent Mapping for Port Translation" on page 279](#)).
- Secured port block allocation (see ["Secured Port Block Allocation for Port Translation" on page 282](#)).

Benefits of Stateful NAT64

Stateful NAT64 provides a way to:

- Let IPv6-only clients contact IPv4 servers using unicast UDP, TCP, or ICMP
- Move to an IPv6 network
- Deal with IPv4 address depletion

RELATED DOCUMENTATION

| [Configuring Next Gen Services Stateful NAT64](#) | 187

IPv4 Addresses Embedded in IPv6 Addresses

Stateful NAT64 and XLAT464 embed IPv4 addresses in IPv6 addresses by using an IPv6 prefix that you specify. The prefix length you use determines how the IPv4 address is embedded.

IPv6 addresses with embedded IPv4 addresses are composed of a variable-length prefix, the embedded IPv4 address, and a variable-length suffix. Bits 64 to 71 are reserved and must be set to 0. The suffix follows the last bit of the embedded IPv4 address, and the suffix bits are ignored and should be set to 0.

The format for the IPv4-embedded IPv6 address depends on the prefix length, as shown in [Table 33 on page 186](#).

Table 33: IPv6 Address With Embedded IPv4 Address

Prefix length	Prefix bits	IPv4 address bits	Reserved bits (must be set to 0)	Suffix bits
32	0-31	32 to 63	64 to 71	72 to 127
40	0 to 39	40 to 63 and 72 to 79	64 to 71	80 to 127
48	0 to 47	48 to 63 and 72 to 87	64 to 71	88 to 127
56	0 to 55	56 to 63 and 72 to 95	64 to 71	96 to 127
64	0 to 63	72 to 103	64 to 71	104 to 127
96	0 to 95	96 to 127	64 to 71	No suffix bits

The following table shows an example of an IPv4 address embedded in an IPv6 address for various prefix lengths.

IPv6 Prefix	IPv4 Address	IPv4 Address Embedded in IPv6 Address
2001:db8::/32	192.0.2.33	2001:db8:c000:221::
2001:db8:100::/40	192.0.2.33	2001:db8:1c0:2:21::
2001:db8:122::/48	192.0.2.33	2001:db8:122:c000:2:2100::
2001:db8:122:300::/56	192.0.2.33	2001:db8:122:3c0:0:221::
2001:db8:122:344::/64	192.0.2.33	2001:db8:122:344:c0:2:2100::
2001:db8:122:344::/96	192.0.2.33	2001:db8:122:344::192.0.2.33

Configuring Next Gen Services Stateful NAT64

IN THIS SECTION

- [Configuring the Source Pool for Stateful NAT64 | 187](#)
- [Configuring the NAT Rules for Stateful NAT64 | 191](#)
- [Configuring the Service Set for Stateful NAT64 | 194](#)
- [Clearing the Don't Fragment Bit | 194](#)

Perform the following steps to configure Next Gen Services Stateful NAT64

Configuring the Source Pool for Stateful NAT64

To configure the source pool for Stateful NAT64:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix to address address-prefix
```

To disable round-robin port allocation for all NAT pools that do not specify an automatic (random-allocation | round-robin) setting, configure the global setting.

```
[edit services nat source]
user@host# set port-round-robin disable
```

3. To configure a range of ports to assign to a pool, perform the following:



NOTE: If you specify a range of ports to assign, the automatic statement is ignored.

- a. Specify the low and high values for the port. If you do not configure automatic port assignment, you must configure a range of ports.

```
[edit services nat source pool nat-pool-name port]
user@host# set range port-low to port-high
```

- b. Specify either random allocation or round-robin allocation. Round-robin allocation is the default.

```
[edit services nat source pool nat-pool-name port range]
user@host# set (random-allocation | round-robin)
```

4. Assign a port within the same range as the incoming port—either 0 through 1023 or 1024 through 65,535. This feature is not available if you configure port-block allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set preserve-range
```

5. Assign a port with the same parity (even or odd) as the incoming port. This feature is not available if you configure port-block allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set preserve-parity
```

6. Configure a global default port range for NAT pools that use port translation. This port range is used when a NAT pool does not specify a port range and does not specify automatic port assignment. The global port range can be from 1024 through 65,535.

```
[edit services nat source]
user@host# set pool-default-port-range port-low to port-high
```

7. Configure the source pool without port translation.

```
[edit services nat source pool nat-pool-name]
user@host# set address-pooling no-paired
```

8. Configure the maximum number of ports that can be allocated for each host. The range is 2 through 65,535.

```
[edit services nat source pool nat-pool-name]
user@host# set limit-ports-per-host number
```

9. If you want to allocate a block of ports for each subscriber to use, configure port-block allocation:
 - a. Configure the number of ports in a block. The range is 1 through 64,512 and the default is 128.

```
[edit services nat source pool nat-pool-name port]
user@host# set block-allocation block-size block-size
```

- b. Configure the interval, in seconds, for which the block is active. After the timeout, a new block is allocated, even if ports are available in the active block. If you set the timeout to 0, port blocks

are filled completely before a new port block is allocated, and the last port block remains active indefinitely. The range is 0 through 86,400, and the default is 0.

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set active-block-timeout timeout-interval
```

- c. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

- d. Configure the maximum number of blocks that can be allocated to a user address. The range is 1 through 512, and the default is 8.

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set maximum-blocks-per-host maximum-block-number
```

- e. Specify how often to send interim system logs for active port blocks and for inactive port blocks with live sessions. This increases the reliability of system logs, which are UDP-based and can get lost in the network. The range is 1800 through 86,400 seconds, and the default is 0 (interim logs are disabled).

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set interim-logging-interval timeout-interval
```

10. Specify the timeout period for endpoint independent translations that use the specified NAT pool. Mappings that are inactive for this amount of time are dropped. The range is 120 through 86,400 seconds. If you do not configure `ei-mapping-timeout`, then the `mapping-timeout` value is used for endpoint independent translations.

```
[edit services nat source pool nat-pool-name]
user@host# set ei-mapping-timeout ei-mapping-timeout
```

11. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

12. To allow the IP addresses of a NAT source pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rules for Stateful NAT64

For Stateful NAT64, you must configure a source rule and a destination rule. To configure the NAT rules for Stateful NAT64:

1. Configure the source NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the IPv6 source addresses that are translated by the NAT rule.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

4. Configure the matching destination address as 0.0.0.0/0.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match destination-address 0.0.0.0/0
```

5. Specify one or more application protocols to which the NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

6. Specify the NAT source pool that contains the addresses for translated source addresses.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

7. Configure endpoint-independent mapping, which ensures that the same external address and port are assigned to all connections from a given host.
 - a. Configure the mapping type as endpoint independent.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set mapping-type endpoint-independent
```

- b. Specify prefix lists that contain the hosts that are allowed to establish inbound connections using the endpoint-independent mapping. (Prefix lists are configured at the [edit policy-options] hierarchy level.)

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set filtering-type endpoint-independent prefix-list [allowed-host] except
[denied-host]
```

- c. Specify the maximum number of inbound flows allowed simultaneously on an endpoint-independent mapping.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set secure-nat-mapping eif-flow-limit number-of-flows
```

- d. Specify the direction in which active endpoint-independent mapping is refreshed. By default, mapping is refreshed for both inbound and outbound active flows.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set secure-nat-mapping mapping-refresh (inbound | inbound-outbound | outbound)
```

8. Configure the destination NAT rule name.

```
[edit services nat destination]
user@host# set rule-set rule-set-name rule rule-name
```

9. Specify the traffic direction to which the destination NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

10. Specify the IPv6 prefix source addresses that are translated by the destination NAT rule. Use the same value that you used for the NAT source rule.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

11. Specify the prefix that is used to embed the IPv4 destination address in the IPv6 destination address.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat destination-prefix destination-prefix
```

12. Configure the IPv6 destination address to match. This is the IPv4 destination address embedded in IPv6 by using the destination-prefix.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

13. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat (source | destination) rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Stateful NAT64

To configure the service set for stateful NAT64:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-interface interface-name
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Clearing the Don't Fragment Bit

To prevent unnecessary creation of IPv6 fragmentation headers when translating IPv4 packets that are less than 1280 bytes, you can specify that the don't fragment (DF) bit for IPv4 packet headers is cleared when the packet length is less than 1280 bytes.

```
[edit services nat natv6v4]
user@host# set clear-dont-fragment-bit
```

RELATED DOCUMENTATION

| [Stateful NAT64 Overview](#) | 185

IPv4 Connectivity Across IPv6-Only Network Using 464XLAT Overview and Configuration

IN THIS CHAPTER

- 464XLAT Overview | 195
- IPv4 Addresses Embedded in IPv6 Addresses | 197
- Configuring 464XLAT Provider-Side Translator for IPv4 Connectivity Across IPv6-Only Network for Next Gen Services | 198

464XLAT Overview

IN THIS SECTION

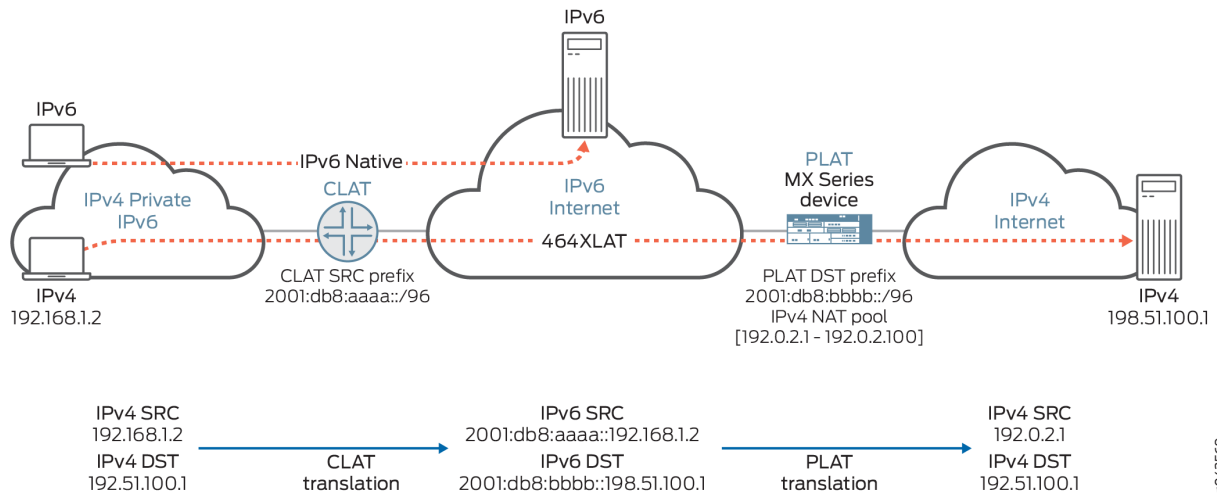
- Benefits of 464XLAT | 197

You can configure the MX Series router as an 464XLAT Provider-Side Translator (PLAT). 464XLAT provides a simple and scalable technique for an IPv4 client with a private address to connect to an IPv4 host over an IPv6 network. 464XLAT only supports IPv4 in the client-server model, so it does not support IPv4 peer-to-peer communication or inbound IPv4 connections.

XLAT464 provides the advantages of not having to maintain an IPv4 network for this IPv4 traffic and not having to assign additional public IPv4 addresses.

A customer-side translator (CLAT), which is not a Juniper Networks product, translates the IPv4 packet to IPv6 by embedding the IPv4 source and destination addresses in IPv6 prefixes, and sends the packet over an IPv6 network to the PLAT. The PLAT translates the packet to IPv4, and sends the packet to the IPv4 host over an IPv4 network (see [Figure 1 on page 196](#)).

Figure 1: 464XLAT Wireline Flow

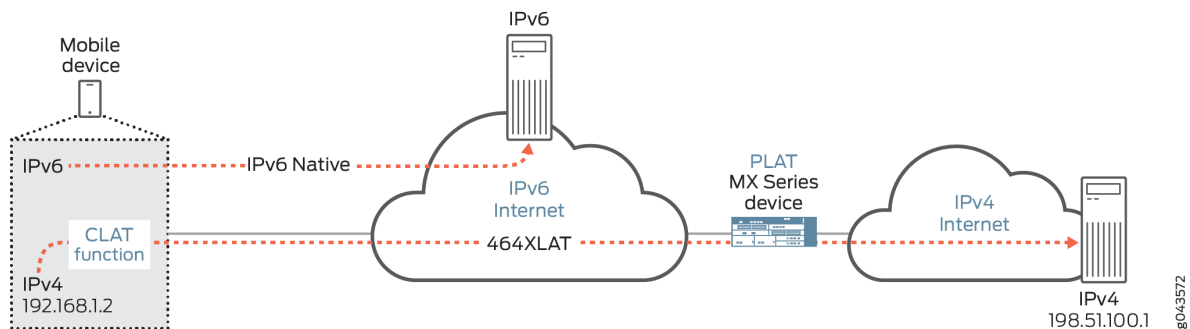


The CLAT uses a unique source IPv6 prefix for each end user, and translates the IPv4 source address to an IPv6 address by embedding it in the IPv6 /96prefix. In [Figure 1 on page 196](#), the CLAT source IPv6 prefix is 2001:db8:aaaa::/96, and the IPv4 source address 192.168.1.2 is translated to 2001:db8:aaaa::192.168.1.2. The CLAT translates the IPv4 destination address to IPv6 by embedding it in the IPv6 prefix of the PLAT (MX Series router). In [Figure 1 on page 196](#), the PLAT destination IPv6 prefix is 2001:db8:bbbb::/96, so the CLAT translates the IPv4 destination address 198.51.100.1 to 2001:db8:bbbb::198.51.100.

The PLAT translates the IPv6 source address to a public IPv4 address, and translates the IPv6 destination address to a public IPv4 address by removing the PLAT prefix.

The CLAT can reside on the end user mobile device in an IPv6-only mobile network, allowing mobile network providers to roll out IPv6 for their users *and* support IPv4-only applications on mobile devices (see [Figure 2 on page 196](#)).

Figure 2: 464XLAT Wireless Flow



464XLAT supports the following:

- Address pooling and endpoint independent mapping (see ["Address Pooling and Endpoint Independent Mapping for Port Translation" on page 279](#)).
- Secured port block allocation (see ["Secured Port Block Allocation for Port Translation" on page 282](#)

Benefits of 464XLAT

- No need to maintain an IPv4 transit network
- No need to assign additional public IPv4 addresses

IPv4 Addresses Embedded in IPv6 Addresses

Stateful NAT64 and XLAT464 embed IPv4 addresses in IPv6 addresses by using an IPv6 prefix that you specify. The prefix length you use determines how the IPv4 address is embedded.

IPv6 addresses with embedded IPv4 addresses are composed of a variable-length prefix, the embedded IPv4 address, and a variable-length suffix. Bits 64 to 71 are reserved and must be set to 0. The suffix follows the last bit of the embedded IPv4 address, and the suffix bits are ignored and should be set to 0.

The format for the IPv4-embedded IPv6 address depends on the prefix length, as shown in [Table 34 on page 197](#).

Table 34: IPv6 Address With Embedded IPv4 Address

Prefix length	Prefix bits	IPv4 address bits	Reserved bits (must be set to 0)	Suffix bits
32	0-31	32 to 63	64 to 71	72 to 127
40	0 to 39	40 to 63 and 72 to 79	64 to 71	80 to 127
48	0 to 47	48 to 63 and 72 to 87	64 to 71	88 to 127
56	0 to 55	56 to 63 and 72 to 95	64 to 71	96 to 127
64	0 to 63	72 to 103	64 to 71	104 to 127

Table 34: IPv6 Address With Embedded IPv4 Address (Continued)

Prefix length	Prefix bits	IPv4 address bits	Reserved bits (must be set to 0)	Suffix bits
96	0 to 95	96 to 127	64 to 71	No suffix bits

The following table shows an example of an IPv4 address embedded in an IPv6 address for various prefix lengths.

IPv6 Prefix	IPv4 Address	IPv4 Address Embedded in IPv6 Address
2001:db8::/32	192.0.2.33	2001:db8:c000:221::
2001:db8:100::/40	192.0.2.33	2001:db8:1c0:2:21::
2001:db8:122::/48	192.0.2.33	2001:db8:122:c000:2:2100::
2001:db8:122:300::/56	192.0.2.33	2001:db8:122:3c0:0:221::
2001:db8:122:344::/64	192.0.2.33	2001:db8:122:344:c0:2:2100::
2001:db8:122:344::/96	192.0.2.33	2001:db8:122:344::192.0.2.33

Configuring 464XLAT Provider-Side Translator for IPv4 Connectivity Across IPv6-Only Network for Next Gen Services

IN THIS SECTION

- [Configuring the Source Pool for 464XLAT | 199](#)
- [Configuring the NAT Rules for 464XLAT | 201](#)
- [Configuring the Service Set for 464XLAT | 204](#)

Configuring the Source Pool for 464XLAT

To configure the source pool for 464XLAT:

1. Create a source NAT pool that is used to translate source IPv6 addresses to source public IPv4 addresses on PLAT.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

3. If you want to allocate a block of ports for each subscriber to use, configure port-block allocation:
 - a. Configure the number of ports in a block. The range is 1 through 64,512 and the default is 128.

```
[edit services nat source pool nat-pool-name port]  
user@host# set block-allocation block-size block-size
```

- b. Configure the interval, in seconds, for which the block is active. After the timeout, a new block is allocated, even if ports are available in the active block. If you set the timeout to 0, port blocks are filled completely before a new port block is allocated, and the last port block remains active indefinitely. The range is 0 through 86,400, and the default is 0.

```
[edit services nat source pool nat-pool-name port block-allocation]  
user@host# set active-block-timeout timeout-interval
```

- c. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

- d. Configure the maximum number of blocks that can be allocated to a user address. The range is 1 through 512, and the default is 8.

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set maximum-blocks-per-host maximum-block-number
```

- e. Specify how often to send interim system logs for active port blocks and for inactive port blocks with live sessions. This increases the reliability of system logs, which are UDP-based and can get lost in the network. The range is 1800 through 86,400 seconds, and the default is 0 (interim logs are disabled).

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set interim-logging-interval timeout-interval
```

4. Specify the timeout period for endpoint independent translations that use the specified NAT pool. Mappings that are inactive for this amount of time are dropped. The range is 120 through 86,400 seconds. If you do not configure `ei-mapping-timeout`, then the `mapping-timeout` value is used for endpoint independent translations.

```
[edit services nat source pool nat-pool-name]
user@host# set ei-mapping-timeout ei-mapping-timeout
```

5. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

Configuring the NAT Rules for 464XLAT

For 464XLAT, you must configure a source rule and a destination rule. To configure the NAT rules for 464XLAT:

1. Configure the source NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the CLAT IPv6 source prefix.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat clat-prefix clat-prefix
```

4. Configure the IPv6 source address prefix to match. This is the IPv4 source address embedded in IPv6 by using the CLAT prefix.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

5. Specify the NAT source pool that the PLAT uses for converting the IPv6 source address to a public IPv4 address.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

6. If you want to ensure that the same external address and port are assigned to all connections from a given host, configure endpoint-independent mapping:

- a. Configure the mapping type as endpoint independent.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set mapping-type endpoint-independent
```

- b. Specify prefix lists that contain the hosts that are allowed to establish inbound connections using the endpoint-independent mapping. (Prefix lists are configured at the [edit policy-options] hierarchy level.)

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set filtering-type endpoint-independent prefix-list [allowed-host] except
[denied-host]
```

- c. Specify the maximum number of inbound flows allowed simultaneously on an endpoint-independent mapping.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set secure-nat-mapping eif-flow-limit number-of-flows
```

- d. Specify the direction in which active endpoint-independent mapping is refreshed. By default, mapping is refreshed for both inbound and outbound active flows.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set secure-nat-mapping mapping-refresh (inbound | inbound-outbound | outbound)
```

- e. Configure the address-pooling paired feature if you want to ensure assignment of the same external IP address for all sessions originating from the same internal host.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat mapping-
type]
user@host# set address-pooling-paired
```

- f. Specify the timeout period for address-pooling-paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

- g. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

7. Configure the destination NAT rule name.

```
[edit services nat destination]
user@host# set rule-set rule-set-name rule rule-name
```

8. Specify the traffic direction to which the destination NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

9. Configure the IPv6 source address prefix to match. Use the same value that you used for the NAT source rule.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

10. Configure the PLAT destination IPv6 prefix.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat destination-prefix address
```


11. Configure the IPv6 destination address to match. This is the IPv4 destination address embedded in IPv6 by using the PLAT destination prefix.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

Configuring the Service Set for 464XLAT

To configure the service set for 464XLAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Clearing the Don't Fragment Bit

Specify that the don't fragment (DF) bit for IPv4 packet headers is cleared when the packet length is less than 1280 bytes.

```
[edit services nat natv6v4]  
user@host# set clear-dont-fragment-bit
```

This prevents unnecessary creation of an IPv6 fragmentation header when translating IPv4 packets that are less than 1280 bytes.

IPv6 NAT Protocol Translation (NAT PT)

IN THIS CHAPTER

- [IPv6 NAT PT Overview | 206](#)
- [IPv6 NAT-PT Communication Overview | 207](#)

IPv6 NAT PT Overview

IPv6 Network Address Translation-Protocol Translation (NAT-PT) provides address allocation and protocol translation between IPv4 and IPv6 addressed network devices. The translation process is based on the Stateless IP/ICMP Translation (SIIT) method; however, the state and the context of each communication are retained during the session lifetime. IPv6 NAT-PT supports Internet Control Message Protocol (ICMP), TCP, and UDP packets.

IPv6 NAT-PT supports the following types of NAT-PT:

- **Traditional NAT-PT**—In traditional NAT-PT, the sessions are unidirectional and outbound from the IPv6 network. Traditional NAT-PT allows hosts within an IPv6 network to access hosts in an IPv4 network. There are two variations to traditional NAT-PT: basic NAT-PT and NAPT-PT.

In basic NAT-PT, a block of IPv4 addresses at an IPv4 interface is set aside for translating addresses as IPv6 hosts as they initiate sessions to the IPv4 hosts. The basic NAT-PT translates the source IP address and related fields such as IP, TCP, UDP, and ICMP header checksums for packets outbound from the IPv6 domain. For inbound packets, it translates the destination IP address and the checksums.

Network Address Port Translation-Protocol Translation (NAPT-PT) can be combined with basic NAT-PT so that a pool of external addresses is used in conjunction with port translation. NAPT-PT allows a set of IPv6 hosts to share a single IPv4 address. NAPT-PT translates the source IP address, source transport identifier, and related fields such as IP, TCP, UDP, and ICMP header checksums, for packets outbound from the IPv6 network. The transport identifier can be a TCP/UDP port or an ICMP query ID. For inbound packets, it translates the destination IP address, destination transport identifier, and the IP and the transport header checksums.

- **Bidirectional NAT-PT**—In bidirectional NAT-PT, sessions can be initiated from hosts in the IPv4 network as well as the IPv6 network. IPv6 network addresses are bound to IPv4 addresses, either statically or dynamically as connections are established in either direction. The static configuration is similar to static NAT translation. Hosts in IPv4 realm access hosts in the IPv6 realm using DNS for address resolution. A DNS ALG must be employed in conjunction with bidirectional NAT-PT to facilitate name-to-address mapping. Specifically, the DNS ALG must be capable of translating IPv6 addresses in DNS queries and responses into their IPv4 address bindings, and vice versa, as DNS packets traverse between IPv6 and IPv4 realms.

The devices partially support the bidirectional NAT-PT specification. It supports flow of bidirectional traffic assuming that there are other ways to convey the mapping between the IPv6 address and the dynamically allocated IPv4 address. For example, a local DNS can be configured with the mapped entries for IPv4 nodes to identify the addresses.

NAT-PT Operation—The devices support the traditional NAT-PT and allow static mapping for the user to communicate from IPv4 to IPv6 . The user needs to statically configure the DNS server with an IPv4 address for the hostname and then create a static NAT on the device for the IPv6-only node to communicate from an IPv4-only node to an IPv6-only node based on the DNS.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.2R1	Starting in Junos OS Release 20.2R1 you can run IPv6 NAT-PT Next Gen Services on MX240, MX480, and MX960 routers.

RELATED DOCUMENTATION

| [NAT46 Next Gen Services Configuration Examples](#)

IPv6 NAT-PT Communication Overview

NAT-PT communication with static mapping— Network Address Translation-Protocol Translation (NAT-PT) can be done in two directions, from IPv6 to IPv4 and vice versa. For each direction, static NAT is used to map the destination host to a local address and a source address NAT is used to translate the source address. There are two types of static NAT and source NAT mapping: one-to-one mapping and prefix-based mapping.

NAT-PT communication with DNS ALG—A DNS-based mechanism dynamically maps IPv6 addresses to IPv4-only servers. NAT-PT uses the DNS ALG to transparently do the translations. For example, a company using an internal IPv6 network needs to be able to communicate with external IPv4 servers that do not yet have IPv6 addresses.

To support the dynamic address binding, a DNS should be used for name resolution. The IPv4 host looks up the name of the IPv6 node in its local configured IPv4 DNS server, which then passes the query to the IPv6 DNS server through a device using NAT-PT.

The DNS ALG in NAT device :

- Translates the IPv6 address resolution back to IPv4 address resolution.
- Allocates an IPv6 address for the mapping.
- Stores a mapping of the allocated IPv4 address to the IPv6 address returned in the IPv6 address resolution so that the session can be established from any-IPv4 hosts to the IPv6 host.

RELATED DOCUMENTATION

| *IPv6 NAT PT Overview*

Stateless Source Network Prefix Translation for IPv6

Overview and Configuration

IN THIS CHAPTER

- [Stateless Source Network Prefix Translation for IPv6 | 209](#)

Stateless Source Network Prefix Translation for IPv6

IN THIS SECTION

- [Stateless Source Network Prefix Translation for IPv6 for IPv6 | 209](#)
- [Configuring NPTv6 for Next Gen Services | 210](#)

Stateless Source Network Prefix Translation for IPv6 for IPv6

IN THIS SECTION

- [Benefits of Stateless Source Network Prefix Translation | 210](#)

When an IPv6 packet is going from an internal network to the external network, Stateless Source Network Prefix Translation for IPv6 (NPTv6) maps the IPv6 prefix of the source address to an IPv6 prefix of an external network. When an IPv6 packet is coming from the external network to the internal network, NPTv6 maps the IPv6 prefix of the destination address to the IPv6 prefix of the internal network.

NPTv6 uses an algorithm to translate the addresses, and does not need to maintain the state for each node or each flow in the translator. NPTv6 also removes the need to recompute the transport layer checksum.

Benefits of Stateless Source Network Prefix Translation

- For edge networks, you do not need to renumber the IPv6 addresses used inside the local network for interfaces, access lists, and system logging messages if:
 - The global prefixes used by the edge network are changed.
 - The IPv6 addresses are used inside the edge network or within other upstream networks (such as multihomed devices) when a site adds, drops, or changes upstream networks.
- IPv6 addresses used by the edge network do not need ingress filtering in upstream networks and do not need their customer-specific prefixes advertised to upstream networks.
- Connections that traverse the translation function are not disrupted by a reset or brief outage of an NPTv6 translator.

Configuring NPTv6 for Next Gen Services

IN THIS SECTION

- [Configuring the Source Pool | 210](#)
- [Configuring the NAT Rule | 211](#)
- [Configuring the Service Set | 212](#)

Configuring the Source Pool

To configure the source pool for NPTv6:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the IPv6 prefix to which the IPv6 source address prefix is translated.

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix
```

Configuring the NAT Rule

To configure the NAT source rule for NPTv6:

1. Configure the NAT rule name.

```
[edit]
user@host# edit services nat source rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the IPv6 prefix of source addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

4. Configure the address-pooling paired feature if you want to ensure assignment of the same external IP address for all sessions originating from the same internal host.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat mapping-type]
user@host# set address-pooling-paired
```

5. Specify the timeout period for address-pooling-paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```


If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

6. Specify the NAT pool that contains the IPv6 prefix for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

7. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set

To configure the service set for NPTv6:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service set, which requires a single service interface, or a next-hop service set, which requires an inside and outside service interface.

- To configure an interface service set:

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface vms-slot-number/pic-number/0.logical-unit-number
```

- To configure a next-hop service set:

```
[edit services service-set service-set-name]
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface vms-slot-number/pic-number/0.logical-unit-number
outside-service-interface vms-slot-number/pic-number/0.logical-unit-number
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]  
user@host# set nat-rule-sets rule-set-name
```

4. Specify that ICMP error messages are sent if NPTv6 address translation fails.

```
[edit services service-set service-set-name nat-options nptv6]  
user@host# set icmpv6-error-messages
```

Transitioning to IPv6 Using Softwires

IN THIS CHAPTER

- [6rd Softwires in Next Gen Services | 214](#)

6rd Softwires in Next Gen Services

IN THIS SECTION

- [6rd Softwires in Next Gen Services Overview | 214](#)
- [Configuring Inline 6rd for Next Gen Services | 215](#)

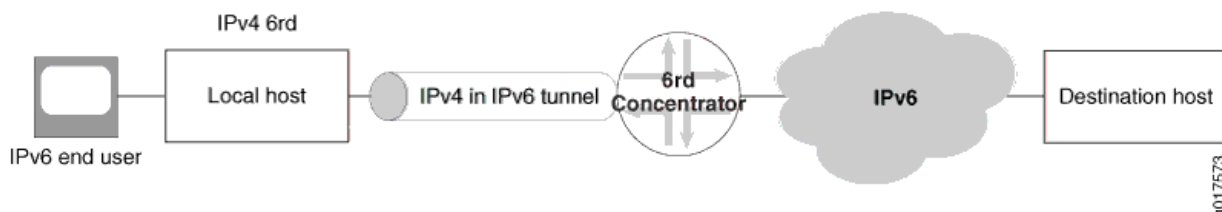
6rd Softwires in Next Gen Services Overview

IN THIS SECTION

- [Benefits | 215](#)

Next Gen Services supports a 6rd softwire concentrator on the MX-SPC3 services card. 6rd softwires allow IPv6 end users to send traffic over an IPv4 network to reach an IPv6 network. IPv6 packets are encapsulated in IPv4 packets by a softwire initiator at the customer edge WAN, and tunneled to a 6rd softwire concentrator. A softwire is created when IPv4 packets containing IPv6 destination information are received at the softwire concentrator, which decapsulates IPv6 packets and forwards them for IPv6 routing.

6rd softwire flow is shown in [Figure 3 on page 215](#).

Figure 3: 6rd Software Flow

In the reverse path, IPv6 packets are sent to the 6rd software concentrator, which encapsulates them in IPv4 packets corresponding to the proper software and sends them to the customer edge WAN.

IPv6 flows are also created for the encapsulated IPv6 payload, and are associated with the specific software that carried them in the first place. When the last IPv6 flow associated with a software ends, the software is deleted. This simplifies configuration and there is no need to create or manage tunnel interfaces.

For more information on 6rd softwires, see RFC 5969, *IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification*.

Benefits

- Rapid deployment of IPv6 service to subscribers on native IPv4 customer edge WANs.
- No need to create or manage tunnel interfaces.

Configuring Inline 6rd for Next Gen Services

IN THIS SECTION

- [Configuring a 6rd Software Concentrator | 215](#)
- [Configuring a 6rd Software Rule | 216](#)
- [Configuring Inline Services and an Inline Services Interface | 217](#)
- [Configuring the IPv4-Facing and IPv6-Facing Interfaces for 6rd | 218](#)
- [Configuring the Service Set | 219](#)

Configuring a 6rd Software Concentrator

To configure a 6rd software concentrator:

1. Configure a 6rd software concentrator name and IP address.

```
user@host# edit services softwires software-name software-name
```

For example:

```
user@host# edit services softwires software-name sw1
```

2. Configure the software type as v6rd and specify a name for it.

```
[edit services softwires software-name sw1]
user@host# set software-type v6rd name
```

For example:

```
[edit services softwires software-name sw1]
user@host# edit software-type v6rd 6rd-sw1
```

3. Configure the 6rd domain's IPv6 prefix.

```
[edit services softwires software-name sw1 software-type v6rd 6rd-sw1]
user@host# set v6rd-prefix v6rd-prefix
```

Configuring a 6rd Software Rule

To configure a 6rd software rule:

1. Specify the name of the rule set that the rule belongs to.

```
[edit services softwires]
user@host# set rule-set rule-set-name
```

2. Specify the direction of traffic to be tunneled.

```
[edit services softwires rule-set rule-set-name]
user@host# set match-direction (input | output)
```

3. Specify the name of the rule.

```
[edit services softwires rule-set rule-set-name]
user@host# set rule rule-name
```

4. Specify the software to apply if the condition is met.

```
[edit services softwires rule-set rule-set-name rule rule-name]
user@host# set then v6rd 6rd-software-name
```

Configuring Inline Services and an Inline Services Interface

Inline services run on MX line cards that can operate under Next Gen Services, for example MPC3 and MPC4 cards. This topic describes how to enable an inline service.

To enable inline services and an inline services interface:

1. Enable inline services for the FPC and PIC slot, and define the amount of bandwidth to dedicate to inline services.

```
[edit chassis fpc slot-number pic number]
user@host# set inline-services bandwidth (1g | 10g | 20g | 30g | 40g | 100g)
```

2. Configure the inline services logical interfaces. Inline interfaces use the following interface naming convention:

```
si-slot/pic/port
```

- If you are using an interface service set, configure one logical unit, and include units for IPv4 and IPv6:

```
user@host# set interfaces si-slot-number/pic-number/0 unit unit-number family inet
user@host# set interfaces si-slot-number/pic-number/0 unit unit-number family inet6
```

For example:

```
user@host# set interfaces si-0/0/0 unit 0 family inet
user@host# set interfaces si-0/0/0 unit 0 family inet6
```

- If you are using a next-hop service set, configure two logical units and define the inside and outside interfaces for IPv4 and IPv6:

```
[edit interfaces si-slot-number/pic-number/0
user@host# set unit inside-unit-number family inet
user@host# set unit inside-unit-number family inet6
user@host# set unit inside-unit-number service-domain inside
user@host# set unit outside-unit-number family inet
user@host# set unit outside-unit-number family inet6
user@host# set unit outside-unit-number service-domain outside
```

For example:

```
user@host# set interfaces si-0/0/0 unit 1 family inet
user@host# set interfaces si-0/0/0 unit 1 family inet6
user@host# set interfaces si-0/0/0 unit 1 service-domain inside
user@host# set interfaces si-0/0/0 unit 2 family inet
user@host# set interfaces si-0/0/0 unit 2 family inet family inet6
user@host# set interfaces si-0/0/0 unit 2 service-domain outside
```

Configuring the IPv4-Facing and IPv6-Facing Interfaces for 6rd

To configure the IPv4-facing and IPv6-facing interfaces:

1. Configure the IPv4-facing interface:

- To configure an interface to use with an interface-style service set, configure input and output service and specify the service set.

```
user@host# set interfaces interface-name unit unit-number family inet service input
service-set service-set-name
user@host# set interfaces interface-name unit unit-number family inet service output
service-set service-set-name
user@host# set interfaces interface-name unit unit-number family inet address ip-address
```

- To configure an interface to use with a next-hop style service set, omit the service input and service output references.

```
user@host# set interfaces interface-name unit unit-number family inet
user@host# set interfaces interface-name unit unit-number family inet address ip-address
```

2. Configure the IPv6-facing interface.

```
user@host# set interface-name unit unit-number family inet6 address ipv6-address
```

Configuring the Service Set

To configure the service set for 6rd processing:

1. Specify a name for the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service set, which requires a single service interface, or a next-hop service set, which requires an inside and outside service interface.

- To configure an interface service set:

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface vms-slot-number/pic-number/0.unit-number
```

- To configure a next-hop service set:

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface vms-slot-number/pic-number/0.inside-unit-number
outside-service-interface vms-slot-number/pic-number/0.outside-unit-number
```

3. Specify the 6rd rule-set that contains the 6rd rule to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set softwires-rule-set software-rule-set-name
```


Transitioning to IPv6 Using DS-Lite Softwires

IN THIS CHAPTER

- [DS-Lite Softwires—IPv4 over IPv6 for Next Gen Services | 220](#)
- [Configuring Next Gen Services DS-Lite Softwires | 223](#)
- [DS-Lite Subnet Limitation | 229](#)
- [Protecting CGN Devices Against Denial of Service \(DOS\) Attacks | 234](#)

DS-Lite Softwires—IPv4 over IPv6 for Next Gen Services

IN THIS SECTION

- [DS-Lite Softwires—IPv4 over IPv6 | 221](#)

Junos OS enables service providers to transition to IPv6 by using software encapsulation and decapsulation techniques. A software is a tunnel that is created between software customer premises equipment (CPE). A software CPE can share a unique common internal state for multiple softwires, making it a very light and scalable solution. When you use softwires, you need not maintain an interface infrastructure for each software, unlike a typical mesh of generic routing encapsulation (GRE) tunnels that requires you to do so. A software initiator at the customer end encapsulates native packets and tunnels them to a software concentrator at the service provider. The software concentrator decapsulates the packets and sends them to their destination. A software is created when a software concentrator receives the first tunneled packet of a flow and prepares the packet for flow processing. The software exists as long as the software concentrator is providing flows for routing. A flow counter is maintained; when the number of active flows is 0, the software is deleted. Statistics are kept for both flows and softwires.

This topic contains the following sections:

DS-Lite Softwires—IPv4 over IPv6

When an ISP begins to allocate new subscriber home IPv6 addresses and IPv6-capable equipment, dual-stack lite (DS-Lite) provides a method for the private IPv4 addresses behind the IPv6 customer edge 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, referred to as a Basic Bridging Broadband (B4), at the customer edge to encapsulate IPv4 packets into IPv6 packets and tunnel them over an IPv6 network to a softwire concentrator, referred to as an Address Family Transition Router (AFTR), for decapsulation. DS-Lite creates the IPv6 softwires that terminate on the services PIC. Packets coming out of the softwire can then have other services such as NAT applied on them.

Starting in Junos OS release 20.2R1, DS-Lite is supported Next Gen Services on MX240, MX480 and MX960 routers with the MX-SPC3.

For more information on DS-Lite softwires, see the IETF draft *Dual Stack Lite Broadband Deployments Following IPv4 Exhaustion*.



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

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

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

DS-Lite and NAT in Next Gen Services

In Next Gen Services, DS-Lite changes the way NAT works with respect to the address-pooling-paired statement for the endpoint independent mapping (EIM), endpoint independent filtering (EIF), and port block allocation (PBA) features. In the earlier Adaptive Services implementation, all of these NAT features are subscriber-based and the subscriber is either a B4 IP address or an IPv6 prefix. In addition, for Adaptive Services, the address-pooling-paired association is between internal IPv4 address and NAT pool address. However in Next Gen Services DS-Lite, the address-pooling-paired pairing is between either the subscriber (B4 IPv6 address or IPv6 prefix) and a NAT pool address. Otherwise, the address-pooling-paired functionality remains the same for Next Gen Services.



NOTE: For CGNAT Next Gen Services on the MX-SPC3 security services card, when you configure DS-Lite use the following rules:

- For non-prefix based DS-Lite subscriber softwires, specify the B4 IPv6 address as the softwire concentrator.
- For prefix-based DS-Lite subscriber softwires, specify the IPv6 prefix address as the softwire concentrator. In addition for prefix-based subscriber DS-Lite softwires, you must specify the subscriber prefix length per service-set under the `[edit softwire-options dslite-ipv6-prefix-length dslite-ipv6-prefix-length]` configuration hierarchy.

You create EIM mappings on a per-softwire basis and they are bound to B4 address; which means the rule matching criteria includes B4 address. For Next Gen Services DS-Lite softwires, there is no special mapping timeout for softwire sessions, instead, they take the value of `inactivity-non-tcp-timeout` as their timeout value.

When a subscriber requires a port to be assigned for the first time, Port Block Allocation (PBA) ensures a block of ports is allocated to that particular subscriber. All subsequent requests from this subscriber use ports from the assigned block. A new port block is allocated when the current active block is exhausted, or after the active port block timeout interval has expired.

DS-Lite and AMS

AMS groups several PICs together and load balances traffic across all PICs that are part of the same group. In a standalone PIC configuration, all softwire sessions originated from any B4, which are destined to a softwire concentrator, are serviced on the same PIC where the softwire concentrator is configured. In the case of a DS-Lite in an AMS configuration, the softwire concentrator is hosted on all PICs in AMS group, however, softwire sessions from various B4 devices are distributed across member PICs. Thus, a softwire session originated from one B4 to the softwire concentrator, is assigned to one member PIC and all packets (IPv4-in-IPv6 and inner IPv4) in both directions (originated from B4 and destined to B4) related to that softwire session are serviced in the same PIC.

For prefix-based DS-Lite subscribers you need to configure the IPv6-prefix for DS-Lite traffic. When a prefix-based subscriber is active, the configured prefix length is taken from the B4 address and is completed with trailing zeros to form a 128-bit IPv6 NAT subscriber. This means that all B4 entities with a matching prefix and all IPv4 networks behind those matching B4 entities, are all identified as a single subscriber. An option is provided to configure the subscriber prefix length per service-set under the `[edit softwire-options dslite-ipv6-prefix-length dslite-ipv6-prefix-length]` hierarchy.



NOTE: For CGNAT Next Gen Services on the MX-SPC3 security services card, when you configure prefix-based DS-Lite subscribers always specify the IPv6 prefix address for the softwire concentrator.

With the prefix-based subscriber feature enabled, only one subscriber context is maintained per-prefix. Hence, the Port Block Allocation (NAT PBA) function would account for port blocks per each subscriber,

instead of every single B4 address. Session limits configured under the software concentrator, limit the number of IPv4 sessions per subscriber, instead of per software/B4 address. Enabling the address-pooling-paired option in prefix-based subscriber configurations results in one public IP address for the subscriber instead of per B4 address.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.2R1	Starting in Junos OS release 20.2R1, DS-Lite is supported Next Gen Services on MX240, MX480 and MX960 routers with the MX-SPC3.

RELATED DOCUMENTATION

<i>Junos Address Aware Network Addressing Overview</i>
Configuring Next Gen Services DS-Lite Softwires 223
<i>DS-Lite Subnet Limitation</i>
<i>DS-Lite Per Subnet Limitation Overview</i>

Configuring Next Gen Services DS-Lite Softwires

IN THIS SECTION

- [Configuring Next Gen Services Software Rules | 223](#)
- [Configuring Service Sets for Next Gen Services Softwires | 225](#)
- [Configuring the DS-Lite Software | 227](#)

Configuring Next Gen Services Software Rules

You configure software rules to instruct the router how to direct traffic to the addresses specified for 6rd, DS-Lite, or MAP-E software concentrators. Software rules do not perform any filtration of the traffic. They do not include a `from` statement, and the only option in the `then` statement is to specify the address of the software concentrator.

Starting in Junos OS release 19.3R2 6rd softwires are supported. Starting in Junos OS release 20.2, DS-Lite and Mapping of Address and Port with Encapsulation (MAP-E).

You can create a software rule consisting of one or more terms and associate a particular 6rd, DS-Lite, or MAP-E software concentrator with each term. You can include the software rule in service sets along with other services rules.

To configure a software rule set:

1. Assign a name to the rule set.

```
[edit services softwires]
user@host# edit rule-set rule-set-name
```

For example:

```
[edit services softwires]
user@host# edit rule-set swrs1
```

2. Configure the input and output match directions for the rule set.

```
[edit services softwires rule-set swrs1]
user@host# set match-direction input
```

3. Specify the name of the rule to apply if the match in this direction is met.

```
[edit services softwires rule-set swrs1]
user@host# edit rule rule-name
```

For example:

```
[edit services softwires rule-set swrs1]
user@host# edit rule swr1
```

4. Associate a 6rd, DS-Lite or MAP-E software concentrator with this term.

```
[edit services softwires rule-set swrs1 rule swr1]
user@host# set then ds-lite | map- | v6rd
```

For example, to associate a DS-Lite software specify the name of the DS-Lite software.

```
[edit services softwares rule-set swrs1 rule swr1]
user@host# set then ds-lite dslsw1
```

5. Repeat steps 2 and 3, and 4 for the output direction.

SEE ALSO

[DS-Lite Softwares—IPv4 over IPv6 for Next Gen Services | 220](#)

DS-Lite Subnet Limitation

DS-Lite Per Subnet Limitation Overview

Configuring Service Sets for Next Gen Services Softwares

You must include previously defined NAT or stateful firewall software rules or a software rule set in a service set to enable software processing.

Starting in Junos OS release 20.2R1, DS-Lite, MAP-E and 6rd softwares are supported in MX240, MX480, and MX960 routers. MAP-E and 6rd softwares are supported inline on an MPC by specifying the si-1/0/0 interface naming convention. DS-Lite is softwares run on the MX-SPC3 security services card.

To configure service sets for softwares:

1. Specify a name for the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

For example:

```
[edit services]
user@host# edit service-set vms-sw-ss
```

2. Specify the IPv6 prefix length for the subscriber addresses.

```
[edit services service-set vms-sw-ss]
user@host# set software-options dslite-ipv6-prefix-length dslite-ipv6-prefix-length
```

We support four prefix lengths: 56, 64, 96 and 128, which is the default.

3. For NAT, you can include a NAT rule for flows originated by DS-Lite softwires.



NOTE:

Currently a NAT rule configuration is required with a DS-Lite software configuration when you use interface service set configurations; NAT is not required when using next-hop service set configurations. NAT processing from IPv4 to IPv6 address pools and vice versa is not currently supported. FTP, HTTP, and RSTP are supported.



NOTE: With a DS-Lite software, if you configure stateful firewall rules without configuring NAT rules, using an interface service set causes the ICMP echo reply messages to not be sent correctly to DS-Lite. This behavior occurs if you apply a service set to both inet and inet6 families. In such a scenario, the traffic that is not destined to the DS-Lite software concentrator is also processed by the service set and the packets might be dropped, although the service set must not process such packets.

To prevent the problem to incorrect processing of traffic applicable for DS-Lite, you must configure a next-hop style service set and not an interface style service set. This problem does not occur when you configure NAT rules with interface service sets for DS-Lite.

Specify the name of the NAT rule set.

```
[edit services service-set vms-sw-ss]
user@host# edit nat-rule-sets nat-rule-set-name
```

4. Specify the service interface to be used.

```
[edit services service-set vms-sw-ss]
user@host# set interface-service service-interface vms-interface-name
```

5. Specify the name of the previously defined softwires rule set that you want to apply to this service set.

```
[edit services service-set vms-sw-ss]
user@host# set softwires-rule-set rule-set-name
```

Configuring the DS-Lite Software

Starting in Junos OS release 20.2R1, you can configure DS-Lite softwires for Next Gen Services on the MX-SPC3 services card.

1. Specify a name for the DS-Lite software.

```
[edit]
user@host# edit services softwires software-types ds-lite name
```

2. Specify a name for the DS-Lite software.

```
[edit]
user@host# edit services softwires software-types ds-lite name
```

For example:

```
user@host# edit services softwires software-types ds-lite dslsw1
```

3. Specify the IPv6 address of the software concentrator.



NOTE: For CGNAT Next Gen Services on the MX-SPC3 security services card, when you configure DS-Lite concentrator, use the following rules:

- For non-prefix based DS-Lite subscribers, specify the B4 IPv6 address
- For prefix-based DS-Lite subscribers, specify the IPv6 prefix address

For example:

```
[edit services softwires software-types ds-lite dslsw1]
user@host# set software-concentrator B4-IPv6-address or IPv6-prefix-address
```

4. You can specify the maximum transmission unit (MTU) for the software tunnel automatically or manually.

- a. To manually specify the MTUs for the software tunnel:

```
[edit services softwires software-types ds-lite dslsw1]
user@host# set mtu-v4 bytes
user@host# set mtu-v6 bytes
```



NOTE: This MTU-v6 option sets the maximum transmission unit when encapsulating IPv4 packets into IPv6. If the final length is greater than the MTU-v4 value, the IPv6 packet is fragmented. This option is mandatory because it depends on other network parameters under administrator control.

5. Specify the maximum number of flows for the software.

```
[edit services softwires software-types ds-lite dslsw1]
user@host# set flow-limit 1000
```

6. (Optional) For prefix-based DS-Lite subscriber softwires, configure the maximum number of subscriber sessions allowed per prefix. You can configure from 0 through 16,384 sessions.

```
[edit services softwires software-types ds-lite dslsw1]
user@host# set session-limit-per-prefix 12
```



NOTE: You cannot use flow-limit and session-limit-per-prefix in the same DS-Lite configuration.

7. Configure the size of the IPv4 subnet prefix to which limiting is applied. ipv4prefix=6rd customer edge ipv4

```
[edit services softwires software-types ds-lite dslsw1]
user@host# set ipv4-prefix
```

8. Configure the size of the IPv6 subnet prefix to which limiting is applied. Specify a prefix length of 56, 64, 96, or 128.

```
[edit services softwires software-types ds-lite dslsw1]
user@host# set v6rd-prefix
```



NOTE: Ensure that all mappings are cleared before changing the prefix length.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.2R1	Starting in Junos OS release 20.2R1, you can configure DS-Lite softwires for Next Gen Services on the MX-SPC3 services card.
20.2R1	Starting in Junos OS release 20.2, DS-Lite and Mapping of Address and Port with Encapsulation (MAP-E).
20.2R1	Starting in Junos OS release 20.2R1, DS-Lite, MAP-E and 6rd softwires are supported in MX240, MX480, and MX960 routers.
19.3R2	Starting in Junos OS release 19.3R2 6rd softwires are supported.

DS-Lite Subnet Limitation

IN THIS SECTION

- [DS-Lite Per Subnet Limitation Overview | 229](#)
- [Configuring DS-Lite Per Subnet Session Limitation to Prevent Denial of Service Attacks | 232](#)

DS-Lite Per Subnet Limitation Overview

Junos OS enables you to limit the number of softwire flows from a subscriber's basic bridging broadband (B4) device at a given point in time, preventing subscribers from excessive use of addresses within the subnet. This limitation reduces the risk of denial-of-service (DoS) attacks. This limitation is supported on MX Series routers equipped with MS-DPCs. Starting in Junos OS Release 18.2R1, MS-MPCs and MS-MICs also support the subnet limitation feature. Starting in Junos OS Release 19.2R1, MX Virtual Chassis and MX Broadband Network Gateway (BNG) routers also support the subnet limitation feature. Starting in Junos OS release 20.2R1, DS-Lite is supported for CGNAT Next Gen Services on MX240, MX480 and MX960 routers.

A household using IPv6 with DS-Lite is a subnet, not just an individual IP address. The subnet limitation feature associates a subscriber and mapping with an IPv6 prefix instead of an IPv6 address. A subscriber can use any IPv6 addresses in that prefix as a DS-Lite B4 address and potentially exhaust carrier-grade NAT resources. The subnet limitation feature enables greater control of resource utilization by identifying a subscriber with a prefix instead of a specific address.

The subnet limit provides the following features:

- Flows utilize the complete B4 address.
- Prefix length can be configured per service set under software-options for the individual service-set.
- Port blocks are allocated per prefix of the subscriber B4 device, and not on each B4 address (if the prefix length is less than 128). If the prefix length is 128, then each IPv6 address is treated as a B4. Port blocks are allocated per 128-bit IPv6 address.
- Session limit, defined under the DS-Lite software concentrator configuration, limits the number of IPv4 sessions for the prefix.
- EIM, EIF, and PCP mappings are created per software tunnel (full 128 bit IPv6 address). Stale mappings time out based on timeout values.
- If prefix length is configured, then PCP max-mappings-per-subscriber (configurable under pcp-server) is based on the prefix only, and not the full B4 address.
- SYSLOGS for PBA allocation and release contain the prefix portion of the address completed with all zeros. SYSLOGS for PCP allocate and release, flow creation and deletion will still contain the complete IPv6 address.

The `show services nat mappings address-pooling-paired` operational command output now shows the mapping for the prefix. The mapping shows the address of the active B4.

The `show services software statistics ds-lite` output includes a new field that displays the number of times the session limit was exceeded for the MPC.

For Next Gen Services on MX240, MX480, and MX960 routers, the subnet limit statistic is displayed in the `Software session limit exceeded` field.

show services software statistics (MX-SPC3)

```
user@host> show services software statistics
vms-2/0/0
  Total Session Interest events      :3
  Total Session Destroy events      :2
  Total Session Public Request events :0
  Total Session Accepts              :1
```

```

Total Session Discards                :0
Total Session Ignores                 :0
Total Session extension alloc failures :0
Total Session extension set failures   :0
Software statistics
Total Software sessions created        :1
Total Software sessions deleted        :2
Total Software sessions created for reverse packets :1
Total Software session create failed for reverse pkts :0
Total Software rule match success      :1
Total Software rule match failed       :0
Software session limit exceeded        :0
Software packet statistics
Total Packets processed                :1
Total packets encapsulated             :1
Total packets decapsulated             :1
Encapsulation errors                   :0
Decapsulation errors                   :0
Encapsulated pkts re-inject failures   :0
Decapsulated pkts re-inject failures   :0
DS-Lite ICMPv4 Echo replies sent      :0
DS-Lite ICMPv4 TTL exceeded messages sent :0
ICMPv6 ECHO request messages received destined to AFTR :0
ICMPv6 ECHO reply messages sent from AFTR :0
ICMPv6 ECHO requests to AFTR process failures :0
V6 untunnelled packets destined to AFTR dropped :1
Software policy add errors              :0
Software policy delete errors          :0
Software policy memory alloc failures   :0
Software Untunnelled packets ignored    :0
Software Misc errors
DS-Lite ICMPv4 TTL exceed message process errors :0

```

SEE ALSO

```

show services nat source mappings address-pooling-paired
show services software statistics

```

Configuring DS-Lite Per Subnet Session Limitation to Prevent Denial of Service Attacks

You can configure the DS-Lite per subnet limitation on MX Series routers equipped with MS-DPCs. Starting in Junos OS Release 18.2R1, MS-MPCs and MS-MICs also support the subnet limitation feature. Starting in Junos OS Release 20.2R1, the Next Gen Services MX-SPC3 security services card supports the subnet limitation feature.

Starting in Junos OS Release 19.2R1, MX Virtual Chassis and MX Broadband Network Gateway (BNG) routers also support the subnet limitation feature.

To configure DS-Lite per subnet session limitation:

1. Configure the size of the subnet prefix to which limiting is applied. Specify a prefix length of 56, 64, 96, or 128.

```
[edit]
user@host# set services service-set service-set-name software-options dslite-ipv6-prefix-length dslite-ipv6-prefix-length
```



NOTE: Ensure that all mappings are cleared before changing the prefix length.

2. If you are using a next-hop service set on an AMS interface for DS-Lite, set the AMS inside interface's IPv6 source prefix length to the same value you use for the subnet prefix in Step 1.


```
[edit interfaces interface-name unit interface-unit-number load-balancing-options hash-keys]
user@host# set ipv6-source-prefix-length ipv6-source-prefix-length
```

3. Configure the maximum number of subscriber sessions allowed per prefix. You can configure from 0 through 16,384 sessions.

```
[edit]
user@host# set services software software-concentrator dslite dslite-concentrator-name session-limit-per-prefix 12
```

For Next Gen Services DS-Lite, MAP-E and V6rd softwires, configure the maximum number of subscriber sessions allowed per prefix:

```
[edit]
user@host# set services softwires software-types ds-lite | map-e | v6rd session-limit-per-prefix limit
```



NOTE: You cannot use flow-limit and session-limit-per-prefix in the same dslite configuration.

SEE ALSO

- clear services nat mappings*
- software-options*
- ds-lite*

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.2R1	Starting in Junos OS release 20.2R1, DS-Lite is supported for CGNAT Next Gen Services on MX240, MX480 and MX960 routers.
20.2R1	Starting in Junos OS Release 20.2R1, the Next Gen Services MX-SPC3 security services card supports the subnet limitation feature.
19.2R1	Starting in Junos OS Release 19.2R1, MX Virtual Chassis and MX Broadband Network Gateway (BNG) routers also support the subnet limitation feature.
18.2R1	Starting in Junos OS Release 18.2R1, MS-MPCs and MS-MICs also support the subnet limitation feature.

Protecting CGN Devices Against Denial of Service (DOS) Attacks

IN THIS SECTION

- [Mapping Refresh Behavior | 234](#)
- [EIF Inbound Flow Limit | 234](#)

You can now choose configuration options that help prevent or minimize the effect of attempted denial of service (DOS) attacks.

Mapping Refresh Behavior

Prior to the implementation of the new options for configuring NAT mapping refresh behavior, described in this topic, a conversation was kept alive when either inbound or outbound flows were active. This remains the default behavior. You can now also specify mapping refresh for only inbound flows or only outbound flows. To configure mapping refresh behavior, include the `mapping-refresh (inbound | outbound | inbound-outbound)` statement at the `[edit services nat rule rule-name term term-name then translated secure-nat-mapping]` hierarchy level.

EIF Inbound Flow Limit

Previously, the number of inbound connections on an EIF mapping was limited only by the maximum flows allowed on the system. You can now configure the number of inbound flows allowed for an EIF. To limit the number of inbound connections on an EIF mapping, include the `eif-flow-limit number-of-flows` statement at the `[edit services nat rule rule-name term term-name then translated secure-nat-mapping]` hierarchy level.

Reducing Traffic and Bandwidth Requirements Using Port Control Protocol

IN THIS CHAPTER

- [Port Control Protocol Overview | 235](#)
- [Configuring Port Control Protocol | 239](#)

Port Control Protocol Overview

IN THIS SECTION

- [Benefits of Port Control Protocol | 237](#)
- [Port Control Protocol Version 2 | 237](#)

Port Control Protocol (PCP) provides a way to control the forwarding of incoming packets by upstream devices, such as NAT44 and firewall devices, and a way to reduce application keepalive traffic. PCP is supported on the MS-DPC, MS-100, MS-400, and MS-500 MultiServices PICs. Starting in Junos OS Release 17.4R1, PCP for NAPT44 is also supported on the MS-MPC and MS-MIC. Starting in Junos OS Release 20.2R1, PCP for CGNAT DS-Lite services are supported for Next Gen Services. Starting in Junos OS Release 18.2R1, PCP on the MS-MPC and MS-MIC supports DS-Lite. In Junos OS Release 18.1 and earlier releases, PCP on the MS-MPC and MS-MIC does not support DS-Lite.

PCP is designed to be implemented in the context of both Carrier-Grade NATs (CGNs) and small NATs (for example, residential NATs). PCP enables hosts to operate servers for a long time (as in the case of a webcam) or a short time (for example, while playing a game or on a phone call) when behind a NAT device, including when behind a CGN operated by their ISP. PCP enables applications to create mappings from an external IP address and port to an internal IP address and port. These mappings are required for successful inbound communications destined to machines located behind a NAT or a firewall. After a mapping for incoming connections is created, remote computers must be informed

about the IP address and port for the incoming connection. This is usually done in an application-specific manner.

Junos OS supports PCP version 2 and version 1.

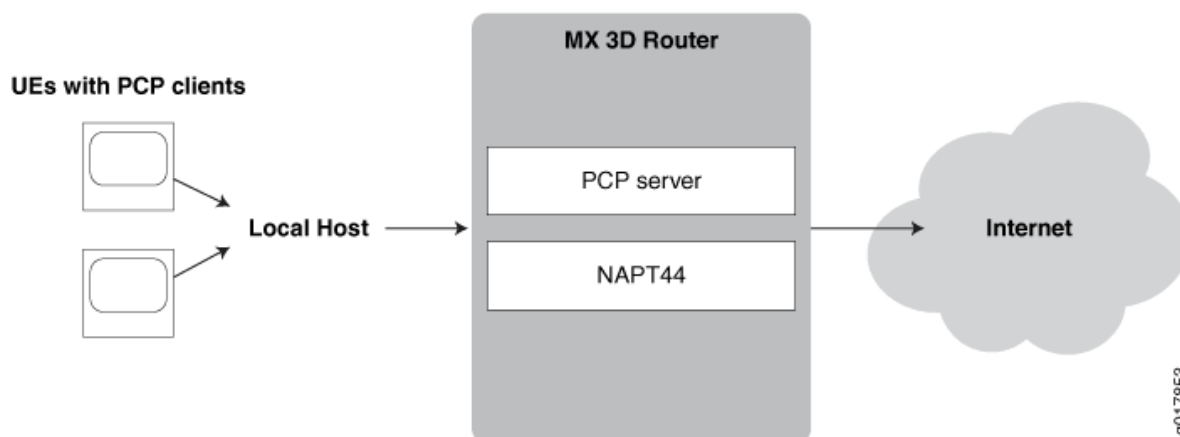
PCP consists of the following components:

- PCP client—A host or gateway that issues PCP requests to a PCP server in order to obtain and control resources.
- PCP server—Typically a CGN gateway or co-located server that receives and processes PCP requests

Junos OS enables configuring PCP servers for mapping flows using NAPT44 capabilities such as port forwarding and port block allocation. Flows can be processed from these sources:

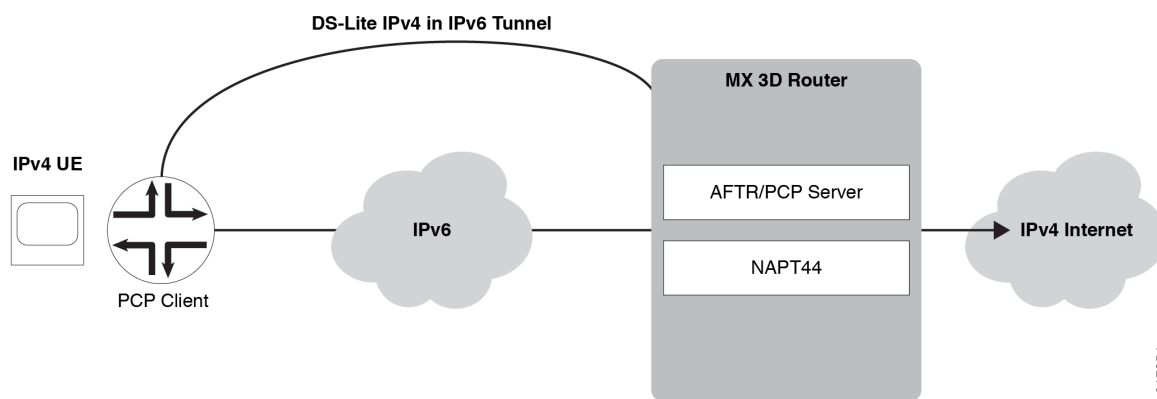
- Traffic containing PCP requests received directly from user equipment, as shown in [Figure 4 on page 236](#).

Figure 4: Basic PCP NAPT44 Topology



- Mapping of traffic containing PCP requests added by a router functioning as a DS-Lite software initiator (B4). This mode, known as *DS-Lite plain mode*, is shown in [Figure 5 on page 237](#).

Figure 5: PCP with DS-Lite Plain Mode



NOTE: Junos OS does not support deterministic port block allocation for PCP-originated traffic.

Benefits of Port Control Protocol

Many NAT-friendly applications send frequent application-level messages to ensure their sessions are not being timed out by a NAT device. PCP is used to:

- Reduce the frequency of these NAT keepalive messages
- Reduce bandwidth on the subscriber's access network
- Reduce traffic to the server
- Reduce battery consumption on mobile devices

Port Control Protocol Version 2

Starting with Junos OS Release 15.1, Port Control Protocol (PCP) version 2 is supported, which is in compliance with RFC 6887. PCP provides a way to control the forwarding of incoming packets by upstream devices, such as NAT44, and firewall devices, and a way to reduce application keep-alive traffic. PCP version 2 supports nonce authentication. PCP allows applications to create mappings from an external IP address and port to an internal IP address and port. A nonce payload prevents a replay attack and it is sent by default unless it is explicitly disabled.

Client nonce verification for version 2 map requests (for refresh or delete) requires that the nonce received in the original map request that causes the PCP mapping to be created is preserved. The version of the initial request that enables the mapping to be created is also preserved. This behavior of

saving the nonce and version parameters denotes that 13 bytes per PCP mapping are used. This slight increase in storage space is not significant when matched with the current memory usage of a system for a single requested mapping (taking into account the endpoint-independent mapping (EIM) and endpoint-independent filtering (EIF) that are created along with it). In a customer deployment, PCP causes EIM and EIF mappings to represent a fraction of all such mappings.

Until Junos Release 15.1, services PICs support PCP servers on Juniper Networks routers in accordance with PCP draft version 22 with version 1 message encoding. With PCP being refined from the draft version as defined in *Port Control Protocol (PCP) draft-ietf-pcp-base-22 (July 2012 expiration)* to a finalized, standard version as defined in RFC 6887 -- Port Control Protocol (PCP), the message encoding changed to version 2 with the addition of a random nonce payload to authenticate peer and map requests as necessary. Version 1 does not decode messages compliant with version 2 format and nonce authentication is not supported. In a real-world network environment, with customer premises equipment (CPE) devices increasingly supporting version 2 only, it is required to parse and send version 2 messages. Backward compatibility with version 1-supporting CPE devices is maintained (version negotiation is part of the standard) and authenticates request nonce payload packets when v2 messages are in use.

The output of the `show services pcp statistics` command contains the PCP unsupported version field, which is incremented to indicate whenever the version is not 1 or 2. A new field, PCP request nonce does not match existing mapping, is introduced to indicate the number of PCP version 2 requests that were ignored because the nonce payload did not match the one recorded in the mapping (authentication failed). If version 2 is in use, the client nonce is used for authentication.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.2R1	Starting in Junos 20.2R1, PCP for CGNAT DS-Lite services are supported for Next Gen Services.
18.2R1	Starting in Junos OS Release 18.2R1, PCP on the MS-MPC and MS-MIC supports DS-Lite.
17.4R1	Starting in Junos OS Release 17.4R1, PCP for NAPT44 is also supported on the MS-MPC and MS-MIC.
15.1	Starting with Junos OS Release 15.1, Port Control Protocol (PCP) version 2 is supported, which is in compliance with RFC 6887.

Configuring Port Control Protocol

IN THIS SECTION

- [Configuring PCP Server Options | 239](#)
- [Configuring a PCP Rule | 241](#)
- [Configuring a NAT Rule | 243](#)
- [Configuring a Service Set to Apply PCP | 243](#)
- [SYSLOG Message Configuration | 244](#)

This topic describes how to configure port control protocol (PCP). PCP is supported on the MS-DPC, MS-100, MS-400, and MS-500 MultiServices PICs. Starting in Junos OS Release 17.4R1, PCP for NAPT44 is also supported on the MS-MPC and MS-MIC. Starting in Junos OS Release 18.2R1, PCP on the MS-MPC and MS-MIC supports DS-Lite. In Junos OS Release 18.1 and earlier releases, PCP on the MS-MPC and MS-MIC does not support DS-Lite. Starting in Junos OS release 20.2R1 PCP is supported on the MX-SPC3 security services card for CGNAT services.

Perform the following configuration tasks:

Configuring PCP Server Options

1. Specify a PCP server name.

```
user @host# edit services pcp server server-name
```

2. Set the IPv4 or IPv6 addresses of the server. For PCP DS-Lite, the *ipv6-address* must match the address of the AFTR (Address Family Transition Router or software concentrator).



NOTE: Starting in Junos OS Release 18.2R1, PCP on the MS-MPC and MS-MIC supports DS-Lite. In Junos OS Release 18.1 and earlier releases, PCP on the MS-MPC and MS-MIC does not support DS-Lite.

```
[edit services pcp server server-name]  
user @host# set ipv6-address ipv6-address
```

or

```
[edit services pcg server server-name]
user @host# set ipv4-address ipv4-address
```

3. For PCP DS-Lite, provide the name of the DS-Lite software concentrator configuration.

```
[edit services pcg server server-name]
user @host# set software-concentrator software-concentrator-name
```

4. Specify the minimum and maximum mapping lifetimes for the server.

```
[edit services pcg server server-name]
user @host# set mapping-lifetime-minimum mapping-lifetime-min
user @host# set mapping-lifetime-maximum mapping-lifetime-max
```

5. Specify the time limits for generating short lifetime or long lifetime errors.

```
[edit services pcg server server-name]
user @host# set short-lifetime-error short-lifetime-error
user @host# set long-lifetime-error long-lifetime-error
```

6. (Optional)—Enable PCP options on the specified PCP server. The following options are available—third-party and prefer-failure. The third-party option is required to enable third-party requests by the PCP client. DS-Lite requires the third-party option. The prefer-failure option requests generation of an error message when the PCP client requests a specific IP address/port that is not available, rather than assigning another available address from the NAT pool. If prefer-failure is not specified NAPT44 assigns an available address/port from the NAT pool based on the configured NAT options.

```
[edit services pcg server server-name]
user @host# set pcg-options third-party
user @host# set pcg-options prefer-failure
```

7. (Optional)—Specify which NAT pool to use for mapping.

```
[edit services pcg server server-name]
user @host# set nat-options pool-name1 <poolName2...>
```



NOTE: When you do not explicitly specify a NAT pool for mapping, the Junos OS performs a partial rule match based on source IP, source port, and protocol, and the Junos OS uses the NAT pool configured for the first matching rule to allocate mappings for PCP.

You *must* use explicit configuration in order to use multiple NAT pools.

For the MX-SPC3 security services card and Next Gen Services, the `nat-options` statement supports only one pool name to attach to a PCP server.

8. (Optional)—Configure the maximum number of mappings per client. The default is 32 and maximum is 128.

```
[edit services pcp server server-name]
user @host# set max-mappings-per-client max-mappings-per-client
```

Configuring a PCP Rule

A PCP rule has the same basic options as all service set rules:

- A `term` option that allows a single rule to have multiple applications.
A term is not required when running the MX-SPC3 security services card for Next Gen Services.
- A `from` option that identifies the traffic that is subject to the rule.
- A `then` option that identifies what action is to be taken. In the case of a PCP rule, this option identifies the PCP server that handles selected traffic

1. Go to the `[edit services pcp rule rule-name]` hierarchy level and specify `match-direction input`.

```
user @host# edit services pcp rule rule-name
user @host# set match-direction input
```

2. Go to the `[edit services pcp rule rule-name term term-name]` hierarchy level and provide a term name.

```
user @host# edit term term-name
```

This step is not required when running the MX-SPC3 security services card for Next Gen Services.

3. (Optional)—Provide a `from` option to filter the traffic to be selected for processing by the rule. When you omit the `from` option, all traffic handled by the service set's service interface is subject to the rule.

The following options are available at the [edit services pcg rule *rule-name* term *term-name* from] hierarchy level:

application-sets <i>set-name</i>	Traffic for the application set is processed by the PCP rule. This step is not required when running the MX-SPC3 security services card for Next Gen Services.
applications [<i>application-name</i>]	Traffic for the application is processed by the PCP rule. This option is not required when running the MX-SPC3 security services card for Next Gen Services.
destination-address <i>address</i> <except>	Traffic for the destination address or prefix is processed by the PCP rule. If you include the except option, traffic for the destination address or prefix is <i>not</i> processed by the PCP rule.
destination-address-range <i>high maximum-value low minimum-value</i> <except>	Traffic for the destination address range is processed by the PCP rule. If you include the except option, traffic for the destination address range is <i>not</i> processed by the PCP rule.
destination-port <i>high maximum-value low minimum-value</i>	Traffic for the destination port range is processed by the PCP rule.
destination-prefix-list <i>list-name</i> <except>	Traffic for a destination address in the prefix list is processed by the PCP rule. If you include the except option, traffic for a destination address in the prefix list is <i>not</i> processed by the PCP rule.
source-address <i>address</i> <except>	Traffic from the source address or prefix is processed by the PCP rule. If you include the except option, traffic from the source address or prefix is <i>not</i> processed by the PCP rule.
source-address-range <i>high maximum-value low minimum-value</i> <except>	Traffic from the source address range is processed by the PCP rule. If you include the except option, traffic from the source address range is <i>not</i> processed by the PCP rule.
source-prefix-list <i>list-name</i> <except>	Traffic from a source address in the prefix list is processed by the PCP rule. If you include the except option, traffic from a source address in the prefix list is <i>not</i> processed by the PCP rule.

4. Set the then option to identify the target PCP server.

```
[edit services pcg rule rule-name term term-name]
user @host# set then pcg-server server-name
```

Configuring a NAT Rule

To configure a NAT rule:

1. Configure the NAT rule name and the match direction.

```
[edit services nat]
user@host# set rule rule-name match-direction match-direction
```

2. Specify the NAT pool to use:

```
[edit services nat rule-name term term-name then translated]
user@host# set source-pool nat-pool-name
```

3. Configure the translation type.

```
[edit services nat rule-name term term-name then translated]
user@host# set translation-type translation-type
```

4. If you are using PCP with IPv4-to-IPv4 NAT or with DS-Lite, configure endpoint-independent mapping (EIM) and endpoint-independent filtering (EIF).

```
[edit services nat rule-name term term-name then translated]
user@host# set mapping-type endpoint-independent
user@host# set filtering-type endpoint-independent
```



NOTE: The PCP mappings are not created if you do not configure EIM and EIF with PCP for IPv4-to-IPv4 NAT or for DS-Lite.

Configuring a Service Set to Apply PCP


To use PCP, you must provide the rule name (or name of a list of rule names) in the `pcp-rule rule-name` option.

1. Go to the `[edit services service-set service-set-name]` hierarchy level.

```
user @host# edit services service-set service-set-name
```


- 2. If this is a new service set, provide basic service set information, including interface information and any other rules that may apply.
- 3. Specify the name of the PCP rule or rule list used to send traffic to the specified PCP server.

```
[edit services service-set service-set-name ]
user @host# set pcp-rule rule-name / rule-listname
```



NOTE: Your service set must also identify any required nat-rule and software-rule.

SYSLOG Message Configuration

A new syslog class, configuration option, `pcp-logs`, has been provided to control PCP log generation. It provides the following levels of logging:

- `protocol`—All logs related to mapping creation, deletion are included at this level of logging.
- `protocol-error`—All protocol error related logs (such as mapping refresh failed, PCP look up failed, mapping creation failed). are included in this level of logging.
- `system-error`—Memory and infrastructure errors are included in this level of logging.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.2R1	Starting in Junos OS release 20.2R1 PCP is supported on the MX-SPC3 security services card for CGNAT services.
18.2R1	
17.4R1	Starting in Junos OS Release 17.4R1, PCP for NAPT44 is also supported on the MS-MPC and MS-MIC.

Transitioning to IPv6 Using MAP-E and MAP-T

IN THIS CHAPTER

- Mapping of Address and Port with Encapsulation (MAP-E) for Next Gen Services | 245
- Equal Cost Multiple Path (ECMP) support for Mapping of Address and Port with Encapsulation (MAP-E) | 252
- Mapping of Address and Port with Translation (MAP-T) for Next Gen Services | 257

Mapping of Address and Port with Encapsulation (MAP-E) for Next Gen Services

IN THIS SECTION

- Understanding Mapping of Address and Port with Encapsulation (MAP-E) | 245
- Configuring Mapping of Address and Port with Encapsulation (MAP-E) for Next Gen Services | 249

Understanding Mapping of Address and Port with Encapsulation (MAP-E)

IN THIS SECTION

- Benefits of Mapping of Address and Port with Encapsulation (MAP-E) | 246
- Mapping of Address and Port with Encapsulation (MAP-E) Terminology | 246
- Mapping of Address and Port with Encapsulation (MAP-E) Functionality | 246
- Mapping of Address and Port with Encapsulation (MAP-E) Supported and Unsupported Features | 247

This topic provides an overview of Mapping of Address and Port with Encapsulation (MAP-E) feature and its benefit to service providers when used as an inline service on MX Series routers with MPC and MIC interfaces. Starting in Junos OS release 20.2R1, MAP-E softwires are supported under Next Gen Services on either an MPC or MIC by specifying the inline services `si-1/1/0` naming convention. Starting in Junos OS release 20.3R1, MPC10E and MX2K-MPC11E support MAP-E.

Benefits of Mapping of Address and Port with Encapsulation (MAP-E)

Reduces administrative overhead and creates a scalable network infrastructure that easily supports connectivity to a large number of IPv4 subscribers over the ISP's IPv6 access network.

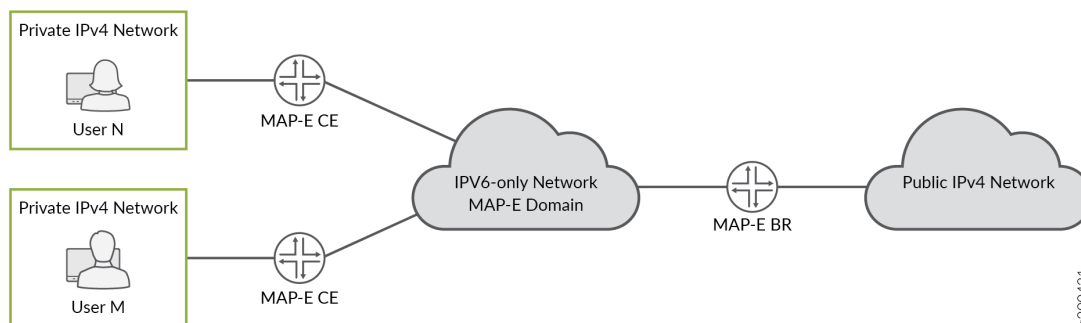
Mapping of Address and Port with Encapsulation (MAP-E) Terminology

1. **Border Relay (BR)**—MAP-E-enabled provider edge device in a MAP domain. A BR device has at least an IPv6-enabled interface and an IPv4 interface connected to the native IPv4 network.
2. **MAP-E Customer Edge (CE)**—MAP-E-enabled customer edge device in a MAP deployment.
3. **MAP domain**—One or more MAP-E CE devices and BR devices connected to the same virtual link.
4. **Port Set ID (PSID)**—Separate part of the transport layer port space that is denoted as port set ID.
5. **Embedded Address (EA) Bits**—EA-bits in the IPv6 address identify an IPv4 prefix or address or a shared IPv4 address and a port-set identifier.
6. **Softwire**—Tunnel between two IPv6 end-points to carry IPv4 packets or two IPv4 end-points to carry IPv6 packets.
7. **Softwire Initiator (SI)**—Softwire at the customer end that encapsulates native packets and tunnels them to a softwire concentrator at the service provider.
8. **Softwire Concentrator (SC)**—Softwire that decapsulates the packets received from a softwire initiator and sends them to their destination.

Mapping of Address and Port with Encapsulation (MAP-E) Functionality

[Figure 6 on page 247](#) illustrates a simple MAP-E deployment scenario.

Figure 6: Sample MAP-E Deployment



In the MAP-E network topology, there are two MAP-E customer edge (CE) devices, each connected to a private IPv4 host. The MAP-E CE devices are dual stack and are capable of Network Address Port Translation (NAPT). The MAP-E CE devices connect to a MAP-E Border Relay (BR) device through an IPv6-only MAP-E network domain. The MAP-E BR device is dual stack and is connected to both a public IPv4 network and an IPv6 MAP-E network.

The MAP-E functionality is as follows:

1. The MAP-E CE devices are capable of NAPT. On receiving an IPv4 packet from the host, the MAP-E CE device performs NAT translation on the incoming IPv4 packets.
2. The NAT translated IPv4 packets are then encapsulated into IPv6 packets by the MAP-E CE device, and sent to the MAP-E BR device.
3. The IPv6 packet gets transported through the IPv6-only service provider network and reaches the MAP-E BR device.
4. On receiving the IPv6 packets, the incoming IPv6 packets are decapsulated by the MAP-E CE device and routed to the IPv4 public network.

In the reverse path, the incoming IPv4 packet is encapsulated into an IPv6 packet by the MAP-E BR device, and routed to the MAP-E CE devices.

Mapping of Address and Port with Encapsulation (MAP-E) Supported and Unsupported Features

Junos OS supports the following MAP-E features and functionality:

- MAP-E implementation supports line card throughput of 100 Gigabits.
- support for Inline MAP-E Border Relay (BR) solution that adheres to draft version 03 of RFC 7597

Fully compliant with draft version 03 of RFC 7597, *Mapping of Address and Port with Encapsulation (MAP)*, when the version-3 option is disabled at the services software-types map-e map-e-concentrator-name

- Support chassis-wide scale of 250 shared MAP-E rules.
- Support the feature on all MPCs using service interfaces with 100 Gigabits.
- Ability to ping MAP-E BR IPv6 address.
- Support only next-hop style of configuration for MAP-E.
- Support reassembly of fragmented IPv4 traffic arriving from IPv4 network before encapsulating it into an IPv6 packet.
- Support fragmentation of inner IPv4 packet if the packet size after encapsulation exceeds the MAP-E maximum transmission unit (MTU).
- Packets having Internet Control Message Protocol (ICMP) payload with the following message types are accepted for MAP-E encapsulation and decapsulation:
 - Echo or Echo Reply Message of type 0 and 8
 - Timestamp or Timestamp Reply Message of type 13 and 14
 - Information Request or Information Reply Message of type 15 and 16
 - Source quench, destination_unreachable, time_exceeded, icmp_redirect, icmp_address_mask_reply and parameter_problem errors
- Border Relay (BR) anycast is supported.

The following features and functionality are not supported with the MAP-E feature:

- Anti-spoof check is not supported for fragmented IPv4 packets coming from a customer edge (CE) device.
- Section 8.2 of the Internet draft draft-ietf-softwire-map-03 (expires on July 28, 2013), *Mapping of Address and Port with Encapsulation (MAP)* is not supported. Instead of responding with an ICMPv6 Destination Unreachable, Source address failed ingress/egress policy (Type 1, Code 5) message, spoof packets are silently dropped and the counter is incremented.
- IPv6 reassembly is not supported.
- ICMP v6-to-v4 translation at the BR is not supported.
- Inline MAP-E with virtual routing and forwarding (VRF) is not supported.
- Inline MAP-E with inline Network Address Translation (NAT) or dual stack (DS)-Lite is not supported.
- Interface-style MAP-E configuration is not supported.

Configuring Mapping of Address and Port with Encapsulation (MAP-E) for Next Gen Services

This example shows you how to configure the MAP-E Border Relay (BR) solution using a next hop-based style of configuration.

To configure MAP-E:

1. Create service interface on the device with 100g bandwidth support.

```
[edit chassis]
user@host# set fpc 0 pic 0 inline-services bandwidth 100g
```

2. Configure the dual stack service interface unit 0.

```
[edit interfaces]
user@host# set si-0/0/0 unit 0 family inet
user@host# set si-0/0/0 unit 0 family inet6
```

3. Configure service interface inside the dual stack domain.

```
[edit interfaces]
user@host# set si-0/0/0 unit 1 family inet
user@host# set si-0/0/0 unit 1 family inet family inet6
user@host# set si-0/0/0 unit 1 service-domain inside
```

4. Configure service interface outside the dual stack domain.

```
[edit interfaces]
user@host# set si-0/0/0 unit 2 family inet
user@host# set si-0/0/0 unit 2 family inet family inet6
user@host# set si-0/0/0 unit 2 service-domain outside
```

5. Configure the IPv4-facing interface on BR.

```
[edit interfaces]
user@host# set ge-0/2/7 unit 0 family inet address 10.10.10.1/16
```

6. Configure the CPE-facing interface on BR.

```
[edit interfaces]
user@host# set ge-0/2/8 unit 0 family inet6 address 3abc::1/16
```

7. Configure the MAP-E software concentrator and associated parameters.

a. (Optional) Configure MAPE version 3.



NOTE: For full RFC 7597 compliance do not configure MAP-E version 3.

b. Specify a name for MAP-E concentrator.

```
[edit]
user@host# edit services softwires software-types map-e mape-tun1
```

c. Specify the IPv6 address of the BR.

```
user@host# set br-address 2001:db8:ffff::1/128
```

d. Specify the rules for the MAP-E concentrator.



NOTE: When configuring the MAP-E software concentrator, take the following into consideration:

- Possible values for ea-bits-len is 0 through 48.
- Possible values for v4-prefix-len is 0 through 32.
- If v4-prefix-len is 0 then ea-bits-len must be non-zero, and vice versa.
- It is possible that ea-bits-len is equal to 0, but psid-len is non-zero.
- If the sum of v4-prefix-len and ea-bits-len is less than 32, then the psid-len must be equal to the difference between 32 and the sum total of v4-prefix-len and ea-bits-len.
- The MAP-E IPv4 and IPv6 prefix must be unique per software concentrator.

- MAP-E PSID offset has a default value of 4, and MAP-E tunnel maximum transmission unit (MTU) has a default value of 9192.

- Specify the rule length for the IPv4 and IPv6 prefixes.

```
user@host# edit services softwires software-types map-e mape-tun1
user@host# edit rule r1
[edit services softwires software-types map-e mape-tun1]
user@host# set rule r1 ipv4-prefix 192.0.2.0/24
user@host# set rule r1 ipv6-prefix 2001:db8:0000::/40
```

- Configure the rule length for embedded addresses.

```
[edit services softwires software-types map-e mape-tun1]
user@host# set ea-bits-length 16
```

- Configure the rule for the PSID offset.

```
[edit services softwires software-types map-e mape-tun1]
user@host# set psid-offset 4
```

- Configure the rule for the PSID length.

```
[edit services softwires software-types map-e mape-tun1]
user@host# set psid-len 8
```

- Specify the MAP-E IPv6 tunnel MTU values.

```
[edit services softwires software-types map-e mape-tun1]
user@host# set mtu-v6 9192
user@host# set v4-reassembly
user@host# set v6-reassembly
```


- vi. Configure the software rule, which specifies the direction of the traffic to be tunneled through the MAP-E software.

```
[edit services softwires]
user@host# set rule-set domain-1 rule r1 then map-e map-e-dom-1
```

8. Configure the service-set for MAP-E.

```
[edit]
user@host# edit services service-set sset1
[edit services service-set sset1]
user@host# set softwires-rule-set domain-1
user@host# set next-hop-service inside-service-interface si-4/2/0.1
user@host# set next-hop-service outside-service-interface si-4/2/0.2
```

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
20.3R1	Starting in Junos OS release 20.3R1, MPC10E and MX2K-MPC11E support MAP-E.
20.2R1	Starting in Junos OS release 20.2R1, MAP-E softwires are supported under Next Gen Services on either an MPC or MIC by specifying the inline services si-1/1/0 naming convention.

Equal Cost Multiple Path (ECMP) support for Mapping of Address and Port with Encapsulation (MAP-E)

IN THIS SECTION

- Equal Cost Multiple Path (ECMP) support for Mapping of Address and Port with Encapsulation (MAP-E) | 253
- Disabling auto-routes to support ECMP with Mapping of Address and Port with Encapsulation (MAP-E) | 253

Equal Cost Multiple Path (ECMP) support for Mapping of Address and Port with Encapsulation (MAP-E)

IN THIS SECTION

- [Benefits](#) | 253

This topic provides an overview of Equal Cost Multiple Path (ECMP) support for Mapping of Address and Port with Encapsulation (MAP-E) feature and its benefit to service providers when used as an inline service on MX Series routers with MPC and MIC interfaces.

In a MAP-E network topology, in the reverse path, the border relay router receives IPv4 traffic and encapsulates it in a IPv6 packet. Longer routes are used for faster matching. However, they do not facilitate EMCP load balancing on the PIC, as the routes point to a single PIC. Starting in 19.3R1, you can disable auto-routes by configuring the `disable-auto-route` statement at the `[edit services software-concentrator map-e <domain-name>]` hierarchy, and direct the static routes to an ECMP load balancer. Hence, the packets can be distributed among different inline service interfaces.

Benefits

Enable load-balancing by distributing packets among different inline service interfaces.

Disabling auto-routes to support ECMP with Mapping of Address and Port with Encapsulation (MAP-E)

This example shows you how to disable auto-routes on a MAP-E Border Relay (BR) solution to support ECMP.

1. Create service interface on the device with 100g bandwidth support.

```
[edit chassis]
user@host# set fpc 0 pic 0 inline-services bandwidth 100g
```

2. Configure the dual stack service interface unit 0.

```
[edit interfaces]
user@host# set si-0/0/0 unit 0 family inet
user@host# set si-0/0/0 unit 0 family inet6
```

3. Configure service interface inside the dual stack domain.

```
[edit interfaces]
user@host# set si-0/0/0 unit 1 family inet
user@host# set si-0/0/0 unit 1 family inet family inet6
user@host# set si-0/0/0 unit 1 service-domain inside
```

4. Configure service interface outside the dual stack domain.

```
[edit interfaces]
user@host# set si-0/0/0 unit 2 family inet
user@host# set si-0/0/0 unit 2 family inet family inet6
user@host# set si-0/0/0 unit 2 service-domain outside
```

5. Configure the IPv4-facing interface on BR.

```
[edit interfaces]
user@host# set ge-0/2/7 unit 0 family inet address 10.10.10.1/16
```

6. Configure the CPE-facing interface on BR.

```
[edit interfaces]
user@host# set ge-0/2/8 unit 0 family inet6 address 3abc::1/16
```

7. Configure MAP-E domain 1 and associated parameters.

```
[edit services software-concentrator]
user@host# set map-e mape-domain-1 version03
user@host# set map-e mape-domain-1 software-address 2001:db8:ffff::1
user@host# set map-e mape-domain-1 ipv4-prefix 192.0.2.0/24 mape-prefix 2001:db8::/32
user@host# set map-e mape-domain-1 ea-bits-len 16
user@host# set map-e mape-domain-1 psid-offset 4
user@host# set map-e mape-domain-1 psid-length 8
user@host# set map-e mape-domain-1 mtu-ipv6 9192
user@host# set map-e mape-domain-1 disable-auto-route
```

8. Configure MAP-E domain 2 and associated parameters.

```
[edit services software software-concentrator]
user@host# set map-e mape-domain-2 version03
user@host# set map-e mape-domain-2 software-address 2001:db8:ffff::1
user@host# set map-e mape-domain-2 ipv4-prefix 192.0.3.0/24 mape-prefix 2002:db8::/32
user@host# set map-e mape-domain-2 ea-bits-len 16
user@host# set map-e mape-domain-2 psid-offset 4
user@host# set map-e mape-domain-2 psid-length 8
user@host# set map-e mape-domain-2 mtu-ipv6 9192
user@host# set map-e mape-domain-2 disable-auto-route
```

9. Configure a software rule for MAP-E domain-1 to specify the direction of traffic to be tunneled.

```
[edit services software]
user@host# set rule sw-rule1 match-direction input term t1 then map-e mape-domain-1
```

10. Configure a software rule for MAP-E domain-2 to specify the direction of traffic to be tunneled.

```
[edit services software]
user@host# set rule sw-rule2 match-direction input term t1 then map-e mape-domain-2
```

11. Configure a single rule-set to combine both the rules.

```
[edit services software]
user@host# set rule-set ecmp-rules rule sw-rule1
user@host# set rule-set ecmp-rules rule sw-rule2
```

12. Configure the service set for MAP-E.

```
[edit services service-set]
user@host# set sset1 software-rule-sets ecmp-rules
user@host# set sset1 next-hop-service inside-service-interface si-0/0/0.1
user@host# set sset1 next-hop-service outside-service-interface si-0/0/0.2
user@host# set sset2 software-rule-sets ecmp-rules
user@host# set sset2 next-hop-service inside-service-interface si-0/1/0.1
user@host# set sset2 next-hop-service outside-service-interface si-0/1/0.2
```

13. Configure static routes for MAP-E BR IPv6 address.

```
[edit routing-options]
user@host# set rib inet6.0 static route 2001:db8:ffff::1/128 next-hop si-0/0/0.1
user@host# set rib inet6.0 static route 2001:db8:ffff::1/128 next-hop si-0/1/0.1
user@host# set rib inet.0 static route 192.0.2.0/24 next-hop si-0/0/0.2
user@host# set rib inet.0 static route 192.0.2.0/24 next-hop si-0/1/0.2
user@host# set rib inet.0 static route 192.0.3.0/24 next-hop si-0/0/0.2
user@host# set rib inet.0 static route 192.0.3.0/24 next-hop si-0/1/0.2
```

14. Enable load balancing.

```
[edit ]
user@host# set policy-options policy-statement LB then load-balance per-packet
user@host# set routing-options forwarding-table export LB
```

15. Verify the status of the routes.

```
[edit ]
user@host# run show route 2001:db8:ffff::1
inet6.0: 10 destinations, 10 routes (10 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2001:db8:ffff::1/128
    *[Static/5] 00:00:12
    > via si-1/0/0.1
    via si-1/1/0.1
```

The service sets of the PICs have *ecmp-rules* configured and they carry the MAP-E rules of domain-1 and domain-2. From the output, you can understand that when the *disable-auto-route* is enabled and *ecmp-rules* configured, instead of the longer auto routes, static routes are created.

RELATED DOCUMENTATION

| *map-e*

Mapping of Address and Port with Translation (MAP-T) for Next Gen Services

IN THIS SECTION

- [Understanding Mapping of Address and Port with Translation \(MAP-T\) | 257](#)
- [Configuring Mapping of Address and Port using Translation \(MAP-T\) | 260](#)

Understanding Mapping of Address and Port with Translation (MAP-T)

IN THIS SECTION

- [Benefits of Mapping of Address and Port with Translation \(MAP-T\) | 257](#)
- [Mapping of Address and Port Using Translation \(MAP-T\) Terminology | 257](#)
- [Mapping of Address and Port Using Translation \(MAP-T\) Functionality | 258](#)
- [Mapping of Address and Port Using Translation \(MAP-T\) Supported and Unsupported Features | 259](#)

This topic provides an overview of Mapping of Address and Port using Translation (MAP-T) feature. The topic lists the benefit to service providers when used as an inline service on MX Series routers with MPC and MIC interfaces.

Benefits of Mapping of Address and Port with Translation (MAP-T)

The translation mode is advantageous in scenarios where the encapsulation overhead, or IPv6 operational practices rule out encapsulation (For example, use of IPv6-only servers, or reliance on IPv6 + protocol headers for traffic classification).

Mapping of Address and Port Using Translation (MAP-T) Terminology

1. **Border Relay (BR)**—MAP-T-enabled provider edge device in a MAP domain. A BR device has at least an IPv6-enabled interface and an IPv4 interface connected to the native IPv4 network.
2. **MAP-T Customer Edge(CE)**—MAP-T-enabled customer edge device in a MAP deployment.
3. **MAP domain**—One or more MAP-T CE devices and BR devices connected to the same virtual link.

4. **Port Set ID (PSID)**—Separate part of the transport layer port space that is denoted as port set ID.
5. **Embedded Address (EA) Bits**—EA-bits in the IPv6 address identify an IPv4 prefix or address or a shared IPv4 address and a port-set identifier.
6. **Softwire**—Tunnel between two IPv6 end-points to carry IPv4 packets or two IPv4 end-points to carry IPv6 packets.
7. **Softwire Initiator (SI)**—Softwire at the customer end that encapsulates native packets and tunnels them to a softwire concentrator at the service provider.
8. **Softwire Concentrator (SC)**—Softwire that decapsulates the packets received from a softwire initiator and sends them to their destination.

Mapping of Address and Port Using Translation (MAP-T) Functionality

The following figure illustrates a simple MAP-T deployment scenario.

Figure 7: Sample MAP-T Deployment

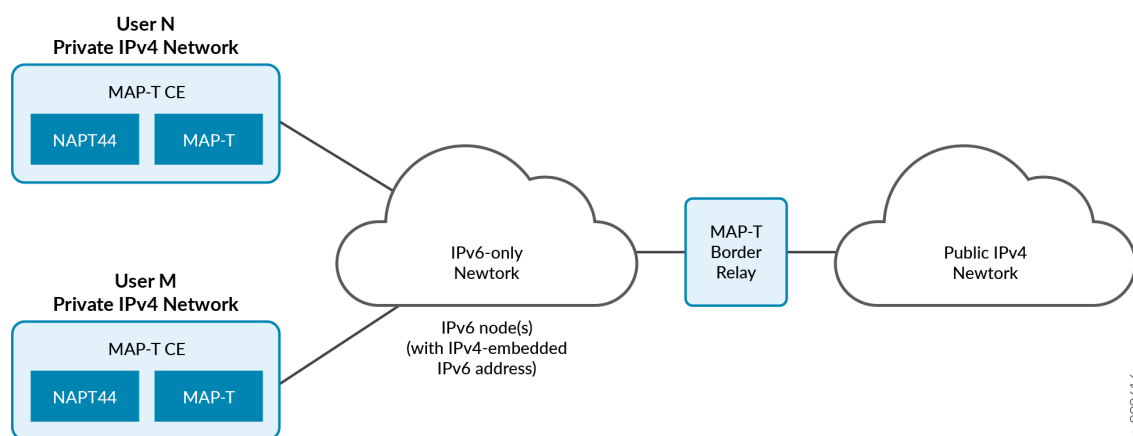


Figure 1 illustrates the MAP-T architecture. It consists of privately addressed IPv4 users (N and M) connected by means of MAP-T CEs to an IPv6 network that is equipped with one or more MAP-T BRs.

The MAP-T CE devices are dual stack and are capable of Network Address Port Translation (NAPT). The MAP-T CE devices connect to a MAP-T Border Relay (BR) device through an IPv6-only MAP-T network domain. The MAP-T BR device is dual stack and is connected to both a public IPv4 network and an IPv6 MAP-T network.

The MAP-T functionality is as follows:

1. The MAP-T CE devices are capable of NAPT. On receiving an IPv4 packet from the host, the MAP-T CE device performs NAT translation on the incoming IPv4 packets. The CE device translate the public IPv4 address and ports into an assigned IPv6 MAP source address. It sends the IPv6 packet with encoded IPv4 information toward the BR.
2. The MAP-T BR checks the source IPv6 MAP address against the configured MAP rules to ensure the correct public IPv4 address and port-range of the CE are encoded in the CE's source IPv6 MAP address. It translates the IPv6 packet into an IPv4 packet and forward it into the public domain.

In the reverse path, the incoming IPv4 packet is translated into an IPv6 packet according to MAP rules. The IPv4 destination address of the received packet is translated into an IPv6 MAP address of the CE.

In full reassembly of packets, the IPv4 fragments from the public IPv4 network are reassembled into a single IPv4 packet which is later translated into IPv6 and routed towards MAP-T CE device. The IPv6 fragments from MAP-T CE device are reassembled into a single IPv6 packet, which are translated to IPv4 packets, and forwarded to the IPv4 cloud.

The following table summarises IPv4 full reassembly capabilities.

Table 35: IPv4/IPv6 Full Reassembly Capabilities

Maximum supported fragments per flow for full reassembly	Maximum IP packet size (in bytes) that can be fully reassembled	Maximum IP fragment size (in bytes)
32	9000	9000

Mapping of Address and Port Using Translation (MAP-T) Supported and Unsupported Features

Junos OS supports the following MAP-T features and functionality:

- TCP/UDP/ICMPv4/ICMPv6 traffic will be forwarded.
- Support only next-hop style of configuration for MAP-T.
- Support maximum packet size of 9192 bytes.
- Support IPv4/IPv6 full reassembly.

The following features and functionality are not supported with the MAP-T feature:

- IPv4/IPv6 partial reassembly is not supported.
- IPv6 fragmentation is not supported.
- Implementation of draft (non-standard) versions of RFC 7599 is not supported.

- Interface-style MAP-T configuration is not supported.
- MAP-T and non-MAP-T service in the same service-set is not supported.

Configuring Mapping of Address and Port using Translation (MAP-T)

This example shows you how to configure the MAP-T Border Relay (BR) solution using a next hop-based style of configuration.

To configure MAP-T:

1. Create service interface on the device with 100g bandwidth support.

```
[edit chassis]
user@host# set fpc 0 pic 0 inline-services bandwidth 100g
```

2. Configure the dual stack service interface unit 0.

```
[edit interfaces]
user@host# set si-0/0/0 unit 0 family inet
user@host# set si-0/0/0 unit 0 family inet6
```

3. Configure service interface inside the dual stack domain.

```
[edit interfaces]
user@host# set si-0/0/0 unit 1 family inet
user@host# set si-0/0/0 unit 1 family inet family inet6
user@host# set si-0/0/0 unit 1 service-domain inside
```

4. Configure service interface outside the dual stack domain.

```
[edit interfaces]
user@host# set si-0/0/0 unit 2 family inet
user@host# set si-0/0/0 unit 2 family inet family inet6
user@host# set si-0/0/0 unit 2 service-domain outside
```

5. Configure the IPv4-facing interface on BR.

```
[edit interfaces]
user@host# set ge-0/2/7 unit 0 family inet address 10.10.10.1/16
```

6. Configure the CPE-facing interface on BR.

```
[edit interfaces]
user@host# set ge-0/2/8 unit 0 family inet6 address 3abc::1/16
```

7. Configure the MAP-T software concentrator and associated parameters.

```
[edit services software software-concentrator]
user@host# set map-t mapt-domain1 dmr-prefix 2001:db8:ffff::/64
user@host# set map-t mapt-domain1 ipv4-prefix 192.0.2.0/24
user@host# set map-t mapt-domain1 mapt-prefix 2001:db8::/32
user@host# set map-t mapt-domain1 ea-bits-len 16
user@host# set map-t mapt-domain1 psid-offset 6
user@host# set map-t mapt-domain1 psid-length 8
user@host# set map-t mapt-domain1 mtu-ipv6 9192
user@host# set map-t mapt-domain1 v4-reassembly
user@host# set map-t mapt-domain1 v6-reassembly
```



NOTE:

- The MAP-T IPv4 prefix must not be 0.0.0.0

8. Configure a software rule to specify the direction of traffic to be tunneled and the map-t software concentrator to be used.

```
[edit services software]
user@host# set rule sw-r1 match-direction input term t1 then map-t mapt-domain1
```

9. Configure the service set for map-t.

```
[edit services service-set]
user@host# set mape-nh-service-set software-rules swire01-r1
user@host# set mape-nh-service-set next-hop-service inside-service-interface si-0/0/0.1
outside-service-interface si-0/0/0.2
```

For example:

```
chassis {
  fpc 4 {
```

```

    pic 0 {
        inline-services {
            bandwidth 100g;
        }
    }
}
fpc 5 {
    pic 0 {
        inline-services {
            bandwidth 100g;
        }
    }
}
}
services {
    service-set sset1 {
        software-rules sw-rule1;
        next-hop-service {
            inside-service-interface si-4/0/0.1;
            outside-service-interface si-4/0/0.2;
        }
    }
    service-set sset2 {
        software-rules sw-rule1;
        next-hop-service {
            inside-service-interface si-5/0/0.1;
            outside-service-interface si-5/0/0.2;
        }
    }
}
software {
    software-concentrator {
        map-t mapt-domain-1 {
            software-address 2001:db8:ffff::1;
            ipv4-prefix 192.0.2.0/24;
            mape-prefix 2001:db8:1234:ab00::/56;
            ea-bits-len 16;
            psid-offset 4;
            psid-length 8;
            mtu-v6 9192;
            version-03;
            v4-reassembly;
            v6-reassembly;
        }
    }
}

```

```

    }
    rule sw-rule1 {
        match-direction input;
        term t1 {
            then {
                map-t mapt-domain-1;
            }
        }
    }
}

interfaces {
    xe-0/1/1 {
        unit 0 {
            family inet6 {
                address 2001:db8::1/32 {
                    ndp 2001:db8:6434:0:00c0:0002:6400:3400 mac 00:11:22:33:44:55;
                }
            }
        }
    }
    xe-0/1/2 {
        unit 0 {
            family inet {
                address 100.1.1.1/24 {
                    arp 100.1.1.2 mac 00:11:22:33:44:55;
                }
            }
        }
    }
    si-4/0/0 {
        unit 1 {
            family inet;
            family inet6;
            service-domain inside;
        }
        unit 2 {
            family inet;
            family inet6;
            service-domain outside;
        }
    }
    si-5/0/0 {

```

```
    unit 1 {  
        family inet6;  
        service-domain inside;  
    }  
    unit 2 {  
        family inet;  
        family inet6;  
        service-domain outside;  
    }  
}  
}
```

Monitoring and Troubleshooting Softwires

IN THIS CHAPTER

- [Ping and Traceroute for DS-Lite | 265](#)
- [Monitoring Softwire Statistics | 266](#)
- [Monitoring CGN, Stateful Firewall, and Softwire Flows | 268](#)

Ping and Traceroute for DS-Lite

With Junos OS Release 11.4, you can use the **ping** and **traceroute** commands to determine the status of the DS-Lite softwire tunnels:

- **IPv6 ping**—The softwire address endpoint on the DS-Lite softwire terminator (AFTR) is usually configured only at the `[edit services softwire]` hierarchy level; it need not be hosted on any interface. Previous releases of the Junos OS software did not provide replies to pings to the IPv6 softwire address when the AFTR was not configured on a specific interface or loopback. An IPv6 ping enables the softwire initiator (B4) to verify the softwire address of the AFTR before creating a tunnel.
- **IPv4 ping**—A special IPv4 address, 192.0.0.1, is reserved for the AFTR. Previous releases of the Junos OS did not respond to any pings sent to this address. A B4 and other IPv4 nodes can now ping to this address to determine whether the DS-Lite tunnel is working.
- **Traceroute**—The AFTR now generates and forwards traceroute packets over the DS-Lite tunnel.



NOTE: No additional CLI configuration is necessary to use the new functionality.

Monitoring Software Statistics

IN THIS SECTION

- [Purpose | 266](#)
- [Action | 266](#)

Purpose

You can review software global statistics by using the **show services software** or `show services software statistics` command.

Action

```
user@host# show services software
Interface: sp-0/0/0, Service set: sset
Software Direction Flow count
2001:0:0:1::1 -> 1001::1 I 3
```

```
user@host# show services software statistics
DS-Lite Statistics:
Service PIC Name: :sp-0/0/0
Statistics
-----
Softwires Created :2
Softwires Deleted :1
Softwires Flows Created :2
Softwires Flows Deleted :1
Slow Path Packets Processed :2
Fast Path Packets Processed :274240
Fast Path Packets Encapsulated :583337
Rule Match Failed :0
Rule Match Succeeded :2
IPv6 Packets Fragmented :0
Transient Errors
-----
```

```

Flow Creation Failed - Retry :0
Slow Path Failed - Retry :0
Errors
-----
Softwire Creation Failed :0
Flow Creation Failed :0
Slow Path Failed :0
Packet not IPv4-in-IPv6 :0
IPv6 Fragmentation Error :0
Slow Path Failed - IPv6 Next Header Offset :0
Decapsulated Packet not IPv4 :0
Fast Path Failed - IPv6 Next Header Offset :0
No Softwire ID :0
No Flow Extension :0
Flow Limit Exceeded :0
6rd Statistics:
Service PIC Name :sp-0/0/0
Statistics
-----
Softwires Created :0
Softwires Deleted :0
Softwires Flows Created :0
Softwires Flows Deleted :0
Slow Path Packets Processed :0
Fast Path Packets Processed :0
Fast Path Packets Encapsulated :0
Rule Match Failed :0
Rule Match Succeeded :0
Transient Errors
-----
Flow Creation Failed - Retry :0
Slow Path Failed - Retry :0
Errors
-----
Softwire Creation Failed :0
Flow Creation Failed :0
Slow Path Failed :0
Packet not IPv6-in-IPv4 :0
Slow Path Failed - IPv6 Next Header Offset :0
Decapsulated Packet not IPv6 :0
Encapsulation Failed - No packet memory :0
No Softwire ID :0

```



```
No Flow Extension :0
ICMPv4 Dropped Packets :0
```

Monitoring CGN, Stateful Firewall, and Softwire Flows

IN THIS SECTION

- [Purpose | 268](#)
- [Action | 268](#)

Purpose

Use the following commands to check the creation of the softwires, pre-NAT flows, and post-NAT flows. Output can be filtered using more specific fields such as AFTR or B4 address or both for DS-Lite, and softwire-concentrator or softwire-initiator or both for 6rd.

- **show services stateful-firewall flows**
- **show services softwire flows**

Action

```
user@host# show services stateful-firewall flows
Interface: sp-0/1/0, Service set: dslite-svc-set2
```

Flow	State	Dir	Frm count
TCP 200.200.200.2:80 -> 44.44.44.1:1025	Forward	0	219942
NAT dest 44.44.44.1:1025 -> 20.20.1.4:1025			
Softwire 2001::2 -> 1001::1			
TCP 20.20.1.2:1025 -> 200.200.200.2:80	Forward	I	110244
NAT source 20.20.1.2:1025 -> 44.44.44.1:1024			
Softwire 2001::2 -> 1001::1			
TCP 200.200.200.2:80 -> 44.44.44.1:1024	Forward	0	219140
NAT dest 44.44.44.1:1024 -> 20.20.1.2:1025			
Softwire 2001::2 -> 1001::1			
DS-LITE 2001::2 -> 1001::1	Forward	I	988729
TCP 200.200.200.2:80 -> 44.44.44.1:1026	Forward	0	218906

	NAT dest	44.44.44.1:1026	->	20.20.1.3:1025		
	Softwire	2001::2	->	1001::1		
TCP		20.20.1.3:1025	->	200.200.200.2:80	Forward I	110303
	NAT source	20.20.1.3:1025	->	44.44.44.1:1026		
	Softwire	2001::2	->	1001::1		
TCP		20.20.1.4:1025	->	200.200.200.2:80	Forward I	110944
	NAT source	20.20.1.4:1025	->	44.44.44.1:1025		
	Softwire	2001::2	->	1001::1		

RELATED DOCUMENTATION

| *Tunneling Services for IPv4-to-IPv6 Transition Overview*

Port Forwarding Overview and Configuration

IN THIS CHAPTER

- [Port Forwarding for Next Gen Services | 270](#)

Port Forwarding for Next Gen Services

IN THIS SECTION

- [Port Forwarding Overview | 270](#)
- [Configuring Port Forwarding with Static Destination Address Translation for Next Gen Services | 271](#)
- [Configuring Port Forwarding without Static Destination Address Translation for Next Gen Services | 275](#)

Port Forwarding Overview

IN THIS SECTION

- [Benefits | 271](#)

Port forwarding allows the public destination address and port of a packet to be translated to an IP address and port in a private network. This translation is a static, one-to-one mapping.

Port forwarding allows a packet to reach a host within a masqueraded, typically private, network, based on the port number on which the packet was received from the originating host. An example of this type of destination is the host of a public HTTP server within a private network.

If you only need to change the destination port, you can also configure port forwarding without translating the destination address.

Port forwarding is supported for destination NAT and twice NAT 44. Port forwarding works only with the FTP application-level gateway (ALG), and has no support for technologies that offer IPv6 services over IPv4 infrastructure, such as IPv6 rapid deployment (6rd) and dual-stack lite (DS-Lite).

Benefits

- Allows remote computers, such as public machines on the Internet, to connect to a non-standard port of a specific computer that is hidden within a private network.

Configuring Port Forwarding with Static Destination Address Translation for Next Gen Services

IN THIS SECTION

- [Configuring the Destination Pool for Destination Address Translation | 271](#)
- [Configuring the Mappings for Port Forwarding | 272](#)
- [Configuring the NAT Rule for Port Forwarding with Destination Address Translation | 272](#)
- [Configuring the Service Set for Port Forwarding with Destination Address Translation | 274](#)

You can configure port forwarding with static destination address translation, which changes the destination address and port of a packet so it can reach the correct host and port within a masqueraded, typically private, network.

Configuring the Destination Pool for Destination Address Translation

To configure the destination pool for the static destination address translation:

1. Create a destination pool.

```
user@host# edit services nat destination pool nat-pool-name
```

2. Define the addresses or subnets to which destination addresses are translated.

```
[edit services nat destination pool nat-pool-name]
user@host# set address address-prefix
```

3. To allow the IP addresses of a NAT destination pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the Mappings for Port Forwarding

1. Configure the port forwarding map name.

```
[edit services nat destination]
user@host# set port-forwarding map-name
```

2. Specify the original destination port number that needs to be translated and the port number to which the original port is mapped. You can configure a maximum of 32 destination port mappings in a port forwarding map.

```
[edit services nat destination port-forwarding map-name]
user@host# set destined-port port-id translated-port port-id
```

In the following example, the destination port number that needs to be translated is 23 and the port to which traffic is mapped is 45.

```
[edit services nat destination port-forwarding map1]
user@host# set destined-port 32 translated-port 45
```

Configuring the NAT Rule for Port Forwarding with Destination Address Translation

To configure the NAT rule for port forwarding with destination address translation:

1. Configure the NAT rule name.

```
[edit services destination source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the destination addresses that the NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

4. Specify the destination port range that the NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-port low-port to high-port
```

5. Specify the NAT pool that contains the destination addresses for translated traffic.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat pool nat-pool-name
```

- Specify the name of the mapping for port forwarding. You can only configure one mapping within a NAT rule term.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then port-forwarding-mappings map-name
```

- Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Port Forwarding with Destination Address Translation

To configure the service set for static destination NAT:

- Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

- Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```



NOTE: You cannot use an AMS interface in a port forwarding service set.

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]  
user@host# set nat-rule-sets rule-set-name
```

Configuring Port Forwarding without Static Destination Address Translation for Next Gen Services

IN THIS SECTION

- [Configuring the Mappings for Port Forwarding | 275](#)
- [Configuring the NAT Rule for Port Forwarding without Destination Address Translation | 276](#)
- [Configuring the Service Set for Port Forwarding without Destination Address Translation | 277](#)

You can configure port forwarding without static destination address translation, which changes the destination port of a packet so it can reach the correct port on the destination host.

Configuring the Mappings for Port Forwarding

1. Configure the port forwarding map name.

```
[edit services destination source]  
user@host# set port-forwarding map-name
```

2. Specify the original destination port number that needs to be translated and the port number to which the original port is mapped. You can configure a maximum of 32 destination port mappings in a port forwarding map.

```
[edit services nat destination port-forwarding map-name]  
user@host# set destined-port port-id translated-port port-id
```


In the following example, the destination port number that needs to be translated is 23 and the port to which traffic is mapped is 45.

```
[edit services nat destination port-forwarding map1]
user@host# set destined-port 32 translated-port 45
```

Configuring the NAT Rule for Port Forwarding without Destination Address Translation

To configure the NAT rule for port forwarding without destination address translation:

1. Configure the NAT rule name.

```
[edit services destination source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the destination addresses that the NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

4. Specify that there is no address translation for the rule.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat off
```

5. Specify the name of the mapping for port forwarding. You can only configure one mapping within a NAT rule term.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then port-forwarding-mappings map-name
```

6. Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Port Forwarding without Destination Address Translation

To configure the service set for static destination NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```



NOTE: You cannot use an AMS interface in a port forwarding service set.

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]  
user@host# set nat-rule-sets rule-set-name
```

Port Translation Features Overview and Configuration

IN THIS CHAPTER

- [Address Pooling and Endpoint Independent Mapping for Port Translation | 279](#)
- [Round-Robin Port Allocation | 281](#)
- [Secured Port Block Allocation for Port Translation | 282](#)

Address Pooling and Endpoint Independent Mapping for Port Translation

IN THIS SECTION

- [Address Pooling | 279](#)
- [Endpoint Independent Mapping and Endpoint Independent Filtering | 280](#)

Address Pooling

Address pooling, or address pooling paired (APP) ensures assignment of the same external IP address for all sessions originating from the same internal host. You can use this feature when assigning external IP addresses from a pool. This option does not affect port utilization.

Address pooling solves the problems of an application opening multiple connections. For example, when Session Initiation Protocol (SIP) client sends Real-Time Transport Protocol (RTP) and Real-Time Control Protocol (RTCP) packets, the SIP generally server requires that they come from the same IP address, even if they have been subject to NAT. If RTP and RTCP IP addresses are different, the receiving endpoint might drop packets. Any point-to-point (P2P) protocol that negotiates ports (assuming address stability) benefits from address pooling paired.

The following are use cases for address pooling:

- A site that offers instant messaging services requires that chat and their control sessions come from the same public source address. When the user signs on to chat, a control session authenticates the user. A different session begins when the user starts a chat session. If the chat session originates from a source address that is different from the authentication session, the instant messaging server rejects the chat session, because it originates from an unauthorized address.
- Certain websites such as online banking sites require that all connections from a given host come from the same IP address.



NOTE: When you deactivate a service set that contains address pooling paired (APP) for that service set, messages are displayed on the PIC console and the mappings are cleared for that service set. These messages are triggered when the deletion of a service-set commences and again generated when the deletion of the service set is completed. The following sample messages are displayed when deletion starts and ends:

- Nov 15 08:33:13.974 LOG: Critical] SVC-SET ss1 (iid 5) deactivate/delete: NAT Mappings and flows deletion initiated
- Nov 15 08:33:14.674 LOG: Critical] SVC-SET ss1 (iid 5) deactivate/delete: NAT Mappings and flows deletion completed

In a scaled environment that contains a large number of APP in a service set, a heavy volume of messages is generated and this process takes some amount of time. We recommend that you wait until the console messages indicating the completion of deletion of the service set are completed before you reactivate the service-set again.

Endpoint Independent Mapping and Endpoint Independent Filtering

Endpoint independent mapping (EIM) ensures the assignment of the same external address *and* port for all connections from a given host if they use the same internal port. This means if they come from a different source port, you are free to assign a different external address.

EIM and APP differ as follows:

- APP ensures assigning the same external IP address.
- EIM provides a stable external IP address and port (for a period of time) to which external hosts can connect. Endpoint independent filtering (EIF) controls which external hosts can connect to an internal host.



NOTE: When you deactivate a service set that contains endpoint independent mapping (EIM) mapping for that service set, messages are displayed on the PIC console and the

mappings are cleared for that service set. These messages are triggered when the deletion of a service set commences and again generated when the deletion of the service set is completed. The following sample messages are displayed when deletion starts and ends:

- Nov 15 08:33:13.974 LOG: Critical] SVC-SET ss1 (iid 5) deactivate/delete: NAT Mappings and flows deletion initiated
- Nov 15 08:33:14.674 LOG: Critical] SVC-SET ss1 (iid 5) deactivate/delete: NAT Mappings and flows deletion completed

In a scaled environment that contains a large number of EIM mappings in a service set, a heavy volume of messages is generated and this process takes some amount of time. We recommend that you wait until the console messages indicating the completion of deletion of the service set are completed before you reactivate the service-set again.

Round-Robin Port Allocation

Round-robin allocation is one method you can configure to allocate private addresses to external addresses and ports. Round-robin allocation assigns one port from each external address in a range before repeating the process for each address in the next range. After ports have been allocated for all addresses in the last range, the allocation process wraps around and allocates the next unused port for addresses in the first range. For example, if you have a NAT pool range of 100.0.0.1 through 100.0.0.12 and the first port is 3333:

- The first connection is allocated to the address:port 100.0.0.1:3333.
- The second connection is allocated to the address:port 100.0.0.2:3333.
- The third connection is allocated to the address:port 100.0.0.3:3333.
- The fourth connection is allocated to the address:port 100.0.0.4:3333.
- The fifth connection is allocated to the address:port 100.0.0.5:3333.
- The sixth connection is allocated to the address:port 100.0.0.6:3333.
- The seventh connection is allocated to the address:port 100.0.0.7:3333.
- The eighth connection is allocated to the address:port 100.0.0.8:3333.
- The ninth connection is allocated to the address:port 100.0.0.9:3333.
- The tenth connection is allocated to the address:port 100.0.0.10:3333.

- The eleventh connection is allocated to the address:port 100.0.0.11:3333.
- The twelfth connection is allocated to the address:port 100.0.0.12:3333.
- Wraparound occurs and the thirteenth connection is allocated to the address:port 100.0.0.1:3334.

Secured Port Block Allocation for Port Translation

You can configure secured port block allocation, which allocates blocks of ports to a subscriber for source NAT port translation. The most recently allocated block is the current active block. New requests for NAT ports for the subscriber are served from the active block. Ports are allocated randomly from the current active block.

Carriers track subscribers using the IP address (RADIUS or DHCP) log. If they use port translation without port block allocation, an IP address is shared by multiple subscribers, and the carrier must track the IP address and port, which are part of the NAT log. Because ports are used and reused at a very high rate, tracking subscribers using the log becomes difficult because of the large number of messages, which are difficult to archive and correlate. By using port block allocation, you can significantly reduce the number of logs, making it easier to track subscribers.

With port block allocation, we generate one syslog log per set of ports allocated for a subscriber. These logs are UDP based and can be lost in the network, particularly for long-running flows. You can configure an interim logging interval to re-send logs for active blocks that have traffic on at least one of the ports.

Static Source NAT Overview and Configuration

IN THIS CHAPTER

- [Static Source NAT Overview | 283](#)
- [Configuring Static Source NAT44 or NAT66 for Next Gen Services | 284](#)

Static Source NAT Overview

IN THIS SECTION

- [Benefits | 283](#)

Static source NAT performs a one-to-one static mapping of the original private domain host source address to a public source address. A block of external addresses is set aside for this mapping, and source addresses are translated as hosts in a private domain originate sessions to the external domain. Static source NAT does not perform port mapping. For packets outbound from the private network, static source NAT translates source IP addresses and related fields such as IP, TCP, UDP, and ICMP header checksums. For inbound packets, static source NAT translates the destination IP address and the checksums.

Benefits

- Allows hosts in the private network to connect with the external domain, while hiding the private network.

Configuring Static Source NAT44 or NAT66 for Next Gen Services

IN THIS SECTION

- [Configuring the Source Pool for Static Source NAT44 or NAT66 | 284](#)
- [Configuring the NAT Rule for Static Source NAT44 or NAT66 | 285](#)
- [Configuring the Service Set for Static Source NAT44 or NAT66 | 286](#)

Configuring the Source Pool for Static Source NAT44 or NAT66

To configure the source pool for static source NAT44 or NAT66:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix to address address-prefix
```

3. Configure a one-to-one static shifting of a range of original source addresses to the range of addresses in the source pool by specifying the base address of the original source address range.

```
[edit services nat source pool nat-pool-name]
user@host# set host-address-base ip-address
```

For example, if the host address base is 198.51.100.30 and the NAT pool uses the range 203.0.113.10 to 203.0.113.20, then 198.51.100.30 translates to 203.0.113.10, 198.51.100.31 translates to 203.0.113.11, and so on.

4. To allow the IP addresses of a NAT source pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rule for Static Source NAT44 or NAT66

To configure the NAT source rule for static source NAT44 or NAT66 :

1. Configure the NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify one or more application protocols to which the NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Specify the NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

6. Configure the address-pooling paired feature if you want to ensure assignment of the same external IP address for all sessions originating from the same internal host.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat mapping-type]
user@host# set address-pooling-paired
```

7. Specify the timeout period for address-pooling-paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

8. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Static Source NAT44 or NAT66

To configure the service set for static source NAT44 or NAT66:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-interface interface-name
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

SEE ALSO

| [Static Source NAT Overview](#) | 283

Static Destination NAT Overview and Configuration

IN THIS CHAPTER

- [Static Destination NAT Overview | 288](#)
- [Configuring Static Destination NAT for Next Gen Services | 289](#)

Static Destination NAT Overview

IN THIS SECTION

- [Benefits of Static Destination NAT | 288](#)

Static destination NAT translates the IPv4 destination address of an incoming packet to the IPv4 address of a private server. This redirects traffic destined to a virtual host (identified by the original destination IP address) to the real host (identified by the translated destination IP address).

Static destination NAT uses a one-to-one mapping between the original address and the translated address; the mapping is configured statically.

You can also statically translate the destination port by using port forwarding. See "[Port Forwarding for Next Gen Services](#)" on page 270.

Benefits of Static Destination NAT

- Allows external traffic to communicate with a private host without revealing the host's private IP address
- Does not require port mapping

RELATED DOCUMENTATION

[Configuring Static Destination NAT for Next Gen Services | 289](#)

Configuring Static Destination NAT for Next Gen Services

IN THIS SECTION

- [Configuring the Destination Pool for Static Destination NAT | 289](#)
- [Configuring the NAT Rule for Static Destination NAT | 289](#)
- [Configuring the Service Set for Static Destination NAT | 291](#)

Configuring the Destination Pool for Static Destination NAT

To configure the destination pool for static destination NAT:

1. Create a destination pool.

```
user@host# edit services nat destination pool nat-pool-name
```

2. Define the addresses or subnets to which destination addresses are translated.

```
[edit services nat destination pool nat-pool-name]  
user@host# set address address-prefix
```

3. To allow the IP addresses of a NAT destination pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]  
user@host# set allow-overlapping-pools
```

Configuring the NAT Rule for Static Destination NAT

To configure the NAT rule for static destination NAT:

1. Configure the NAT rule name.

```
[edit services nat destination]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the destination NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the source addresses of traffic that the NAT rule applies to.

To specify one address or prefix value:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify the destination addresses that the NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

5. Specify one or more application protocols to which the destination NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

6. Specify the NAT pool that contains the destination addresses for translated traffic.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat pool nat-pool-name
```

7. Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Static Destination NAT

To configure the service set for static destination NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```


2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]  
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]  
user@host# set next-hop-service inside-service-interface interface-name outside-service-  
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]  
user@host# set nat-rule-sets rule-set-name
```

RELATED DOCUMENTATION

| [Static Destination NAT Overview](#) | 288

Twice NAT Overview and Configuration

IN THIS CHAPTER

- [Twice NAT Overview | 293](#)
- [Configuring Twice NAT for Next Gen Services | 294](#)

Twice NAT Overview

IN THIS SECTION

- [Benefits | 293](#)

Twice NAT translates both the source and destination IP addresses.

The private source address is translated by dynamically assigning a public address from a pool and a port number. Multiple private IP addresses can be mapped to the same external address because each private address is mapped to a different port of the external address.

The destination address is translated to the IPv4 address of a private server. This redirects traffic destined to a virtual host (identified by the original destination IP address) to the real host (identified by the translated destination IP address). The destination addresses is translated with a one-to-one static mapping to an address in a pool. Port mapping is not performed for the destination address.

You can also statically translate the destination port by using port forwarding. See "[Port Forwarding for Next Gen Services](#)" on page 270.

Benefits

- Allows hosts in the private network to connect with the external domain, while hiding the private network.

- Minimizes the number of public IP addresses that are allocated for NAT.
- Allows external traffic to communicate with a private host without revealing the host's private IP address

Configuring Twice NAT for Next Gen Services

IN THIS SECTION

- [Configuring the Source and Destination Pools for Twice NAT | 294](#)
- [Configuring the NAT Rules for Twice NAT | 298](#)
- [Configuring the Service Set for Twice NAT | 301](#)

Configuring the Source and Destination Pools for Twice NAT

To configure the source and destination pools for twice NAT:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix to address address-prefix
```

3. To configure automatic port assignment, specify either random allocation or round-robin allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set automatic (random-allocation | round-robin)
```

Random allocation randomly assigns a port from the range 1024 through 65535 for each port translation. Round robin allocation first assigns port 1024, and uses the next higher port for each successive port assignment. Round robin allocation is the default.

4. To disable round-robin port allocation for all NAT pools that do not specify an automatic (random-allocation | round-robin) setting, configure the global setting.

```
[edit services nat source]
user@host# set port-round-robin disable
```

5. To configure a range of ports to assign to a pool, perform the following:



NOTE: If you specify a range of ports to assign, the automatic statement is ignored.

- a. Specify the low and high values for the port. If you do not configure automatic port assignment, you must configure a range of ports.

```
[edit services nat source pool nat-pool-name port]
user@host# set range port-low to port-high
```

- b. Specify either random allocation or round-robin allocation. Round-robin allocation is the default.

```
[edit services nat source pool nat-pool-name port range]
user@host# set (random-allocation | round-robin)
```

6. Assign a port within the same range as the incoming port—either 0 through 1023 or 1024 through 65,535. This feature is not available if you configure port-block allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set preserve-range
```

7. Assign a port with the same parity (even or odd) as the incoming port. This feature is not available if you configure port-block allocation.

```
[edit services nat source pool nat-pool-name port]
user@host# set preserve-parity
```

8. Configure a global default port range for NAT pools that use port translation. This port range is used when a NAT pool does not specify a port range and does not specify automatic port assignment. The global port range can be from 1024 through 65,535.

```
[edit services nat source]
user@host# set pool-default-port-range port-low to port-high
```

9. If you want to allocate a block of ports for each subscriber to use for NAPT, configure port-block allocation:
 - a. Configure the number of ports in a block. The range is 1 through 64,512 and the default is 128.

```
[edit services nat source pool nat-pool-name port]
user@host# set block-allocation block-size block-size
```

- b. Configure the interval, in seconds, for which the block is active. After the timeout, a new block is allocated, even if ports are available in the active block. If you set the timeout to 0, port blocks are filled completely before a new port block is allocated, and the last port block remains active indefinitely. The range is 0 through 86,400, and the default is 0.

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set active-block-timeout timeout-interval
```

- c. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

- d. Configure the maximum number of blocks that can be allocated to a user address. The range is 1 through 512, and the default is 8.

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set maximum-blocks-per-host maximum-block-number
```

- e. Specify how often to send interim system logs for active port blocks and for inactive port blocks with live sessions. This increases the reliability of system logs, which are UDP-based and can get lost in the network. The range is 1800 through 86,400 seconds, and the default is 0 (interim logs are disabled).

```
[edit services nat source pool nat-pool-name port block-allocation]
user@host# set interim-logging-interval timeout-interval
```

10. Specify the timeout period for endpoint independent translations that use the specified NAT pool. Mappings that are inactive for this amount of time are dropped. The range is 120 through 86,400 seconds. If you do not configure `ei-mapping-timeout`, then the `mapping-timeout` value is used for endpoint independent translations.

```
[edit services nat source pool nat-pool-name]
user@host# set ei-mapping-timeout ei-mapping-timeout
```

11. Specify the timeout period for address-pooling paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

12. Define the NAT pool utilization levels that trigger SNMP traps. The `raise-threshold` is the pool utilization percentage that triggers the trap, and the range is 50 through 100. The `clear-threshold` is the pool utilization percentage that clears the trap, and the range is 40 through 100. For pools that use port-block allocation, the utilization is based on the number of ports that are used; for pools

that do not use port-block allocation, the utilization is based on the number of addresses that are used.

```
[edit services nat source pool nat-pool-name]
user@host# set pool-utilization-alarm raise-threshold value
user@host# set pool-utilization-alarm clear-threshold value
```

If you do not configure `pool-utilization-alarm`, traps are not created.

13. Create a destination pool. Do not use the same name that you used for the source pool.

```
user@host# edit services nat destination pool nat-pool-name
```

14. Define the addresses or subnets to which destination addresses are translated.

```
[edit services nat destination pool nat-pool-name]
user@host# set address address-prefix
```

15. To allow the IP addresses of a NAT source pool or destination pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`. However, pools that configure port-block allocation must not overlap with other pools.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rules for Twice NAPT

To configure the source and destination NAT rules for twice NAPT:

1. Configure the source NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify one or more application protocols to which the NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Specify the NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

6. If you want to ensure that the same external address and port are assigned to all connections from a given host, configure endpoint-independent mapping:
 - a. Configure the mapping type as endpoint independent.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set mapping-type endpoint-independent
```


- b. Specify prefix lists that contain the hosts that are allowed to establish inbound connections using the endpoint-independent mapping. (Prefix lists are configured at the [edit policy-options] hierarchy level.)

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set filtering-type endpoint-independent prefix-list [allowed-host] except
[denied-host]
```

- c. Specify the maximum number of inbound flows allowed simultaneously on an endpoint-independent mapping.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set secure-nat-mapping eif-flow-limit number-of-flows
```

- d. Specify the direction in which active endpoint-independent mapping is refreshed. By default, mapping is refreshed for both inbound and outbound active flows.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat]
user@host# set secure-nat-mapping mapping-refresh (inbound | inbound-outbound | outbound)
```

7. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

8. Configure the destination NAT rule name.

```
[edit services nat destination]
user@host# set rule-set rule-set-name rule rule-name
```

9. Specify the traffic direction to which the destination NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

10. Specify the destination addresses of traffic that the destination NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

11. Specify one or more application protocols to which the destination NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

12. Specify the destination NAT pool that contains the destination addresses for translated traffic.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat pool nat-pool-name
```

13. Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Twice NAT

To configure the service set for twice NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set. Include the source NAT rule set and the destination NAT rule set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Twice NAT Overview and Configuration

IN THIS CHAPTER

- [Twice Static NAT Overview | 303](#)
- [Configuring Twice Static NAT44 for Next Gen Services | 304](#)
- [Twice Dynamic NAT Overview | 309](#)
- [Configuring Twice Dynamic NAT for Next Gen Services | 309](#)

Twice Static NAT Overview

IN THIS SECTION

- [Benefits | 303](#)

Twice static NAT translates both the source and destination IP addresses. An addresses is translated with a one-to-one static mapping to an address in a pool. Port mapping is not performed.

The original private domain host source address is translated to a public source address.

The destination address is translated to the IPv4 address of a private server. This redirects traffic destined to a virtual host (identified by the original destination IP address) to the real host (identified by the translated destination IP address).

Benefits

- Allows hosts in the private network to connect with the external domain, while hiding the private network.
- Hides a private network

- Allows external traffic to communicate with a private host without revealing the host's private IP address
- Does not require port mapping

Configuring Twice Static NAT44 for Next Gen Services

IN THIS SECTION

- [Configuring the Source and Destination Pools for Twice Static NAT44 | 304](#)
- [Configuring the NAT Rules for Twice Static NAT44 | 305](#)
- [Configuring the Service Set for Twice Static NAT44 | 308](#)

Configuring the Source and Destination Pools for Twice Static NAT44

To configure the source and destination pools for twice static NAT44:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix to address address-prefix
```

3. Configure a one-to-one static shifting of a range of original source addresses to the range of addresses in the source pool by specifying the base address of the original source address range.

```
[edit services nat source pool nat-pool-name]
user@host# set host-address-base ip-address
```

For example, if the host address base is 198.51.100.30 and the NAT pool uses the range 203.0.113.10 to 203.0.113.20, then 198.51.100.30 translates to 203.0.113.10, 198.51.100.31 translates to 203.0.113.11, and so on.

4. Create a destination pool. Do not use the same name that you used for the source pool.

```
user@host# edit services nat destination pool nat-pool-name
```

5. Define the addresses or subnets to which destination addresses are translated.

```
[edit services nat destination pool nat-pool-name]
user@host# set address address-prefix
```

6. To allow the IP addresses of a NAT pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rules for Twice Static NAT44

To configure the source and destination NAT rules for twice static NAT44:

1. Configure the source NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify one or more application protocols to which the source NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Specify the source NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

6. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

7. Configure the destination NAT rule name.

```
[edit services nat destination]
user@host# set rule-set rule-set-name rule rule-name
```

8. Specify the traffic direction to which the destination NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

9. Specify the destination addresses of traffic that the destination NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

10. Specify one or more application protocols to which the destination NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

11. Specify the destination NAT pool that contains the destination addresses for translated traffic.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat pool nat-pool-name
```


12. Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Twice Static NAT44

To configure the service set for twice static NAT44:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set. Include the source NAT rule set and the destination NAT rule set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Twice Dynamic NAT Overview

IN THIS SECTION

- [Benefits | 309](#)

Twice dynamic NAT translates both the source and destination IP addresses. Port mapping is not performed.

The private source address is translated by dynamically assigning a public address from a pool, and the mapping from the original source address to the translated source address is maintained as long as there is at least one active flow that uses this mapping.

The destination address is translated to the IPv4 address of a private server. This redirects traffic destined to a virtual host (identified by the original destination IP address) to the real host (identified by the translated destination IP address). The destination addresses is translated with a one-to-one static mapping to an address in a pool.

Benefits

- Allows hosts in the private network to connect with the external domain, while hiding the private network.
- Allows a few public IP addresses to be used by several private hosts
- Allows external traffic to communicate with a private host without revealing the host's private IP address
- Does not require port mapping

Configuring Twice Dynamic NAT for Next Gen Services

IN THIS SECTION

- [Configuring the Source and Destination Pools for Twice Dynamic NAT | 310](#)
- [Configuring the NAT Rules for Twice Dynamic NAT | 311](#)

- [Configuring the Service Set for Twice Dynamic NAT | 314](#)

Configuring the Source and Destination Pools for Twice Dynamic NAT

To configure the source and destination pools for twice dynamic NAT:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix to address address-prefix
```

3. Disable port translation.

```
[edit services nat destination pool nat-pool-name]  
user@host# set port no-translation
```

4. Define the NAT pool utilization levels that trigger SNMP traps. The raise-threshold is the pool utilization percentage that triggers the trap, and the range is 50 through 100. The clear-threshold is the pool utilization percentage that clears the trap, and the range is 40 through 100. The utilization is based on the number of addresses that are used.

```
[edit services nat source pool nat-pool-name]  
user@host# set pool-utilization-alarm raise-threshold value  
user@host# set pool-utilization-alarm clear-threshold value
```

If you do not configure pool-utilization-alarm, traps are not created.

5. Create a destination pool. Do not use the same name that you used for the source pool.

```
user@host# edit services nat destination pool nat-pool-name
```

6. Define the addresses or subnets to which destination addresses are translated.

```
[edit services nat destination pool nat-pool-name]
user@host# set address address-prefix
```

7. To allow the IP addresses of a NAT source pool or destination pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rules for Twice Dynamic NAT

To configure the source and destination NAT rules for twice dynamic NAT:

1. Configure the source NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

To specify any unicast address:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address any-unicast
```

4. Specify one or more application protocols to which the source NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

5. Configure the address-pooling paired feature if you want to ensure assignment of the same external IP address for all sessions originating from the same internal host.

```
[edit services nat source rule-set rule-set-name rule rule-name then source-nat mapping-type]
user@host# set address-pooling-paired
```

6. Specify the timeout period for address-pooling-paired mappings that use the NAT pool. The range is 120 through 86,400 seconds, and the default is 300. Mappings that are inactive for this amount of time are dropped.

```
[edit services nat source pool nat-pool-name]
user@host# set mapping-timeout mapping-timeout
```

If you do not configure `ei-mapping-timeout` for endpoint independent translations, then the `mapping-timeout` value is used for endpoint independent translations.

7. Specify the source NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

8. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

9. Configure the destination NAT rule name.

```
[edit services nat destination]
user@host# set rule-set rule-set-name rule rule-name
```

10. Specify the traffic direction to which the destination NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

11. Specify the destination addresses of traffic that the destination NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

12. Specify one or more application protocols to which the destination NAT rule applies. The number of applications listed in the rule must not exceed 3072.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match application [application-name]
```

13. Specify the destination NAT pool that contains the destination addresses for translated traffic.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat pool nat-pool-name
```

14. Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Twice Dynamic NAT

To configure the service set for twice dynamic NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service, which requires a single service interface, or a next-hop service, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the NAT rule sets to be used with the service set. Include the source NAT rule set and the destination NAT rule set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Class of Service Overview and Configuration

IN THIS CHAPTER

- [Class of Service for Services PICs \(Next Gen Services\) | 315](#)

Class of Service for Services PICs (Next Gen Services)

IN THIS SECTION

- [Class of Service Overview for Services PICs \(Next Gen Services\) | 315](#)
- [Configuring CoS for Traffic Processed by a Services PIC \(Next Gen Services\) | 316](#)

Class of Service Overview for Services PICs (Next Gen Services)

IN THIS SECTION

- [Benefits | 316](#)

You can configure CoS Differentiated Services (DiffServ) code point (DSCP) marking and forwarding-class assignment for packets transiting a services PIC while being processed by a service set.

Configure services CoS rules, which identify the matching conditions for packet source and destination addresses and for packet applications, and the actions to take on those packets. You must apply CoS rules to a service set before the rules can be applied to traffic. Only stateful firewall and NAT rules can be used with CoS rules in a service set.

You can also configure specific CoS actions for FTP and for SIP traffic by creating an application profile. The application profile can then be referenced in the CoS rule actions.

The services CoS rules do not support scheduling. You must configure scheduling at the [edit class-of-service] hierarchy level on the output interface or fabric.



NOTE: When configuring Next Gen Services with the MX-SPC3 services card, the service set must include at least one stateful firewall (SFW) rule or NAT rule, or services CoS does not work. Only stateful firewall and NAT rules can be used with CoS rules in a service set. CoS works without NAT and SFW rules also.

Benefits

CoS for traffic on a services PIC lets you classify traffic flows based on stateful firewall and NAT configurations.

SEE ALSO

Configuring CoS for Traffic Processed by a Services PIC (Next Gen Services)

Configuring CoS for Traffic Processed by a Services PIC (Next Gen Services)

IN THIS SECTION

- [Configuring CoS Rules | 316](#)
- [Configuring Application Profiles for CoS Rules | 319](#)
- [Configuring CoS Rule Sets | 321](#)
- [Configuring the Service Set for CoS | 321](#)

Configuring CoS Rules

1. Configure a name for the CoS rule.

```
user@host# edit services cos rule rule-name
```

2. Specify the traffic flow direction for the CoS rule.

```
[edit services cos rule rule-name]
user@host# set match-direction (input | input-output | output)
```

If this CoS rule is applied to an interface-type service set, the direction is determined by whether a packet is entering or leaving the interface on which the service set is applied. If this CoS rule is applied to a next-hop service set, the direction is input if the inside interface is used to route the packet, and the direction is output if the outside interface is used to route the package.

If you configure input-output, the rule is applied to sessions initiated from either direction.

3. Configure a name for a CoS rule policy.

```
[edit services cos rule rule-name]
user@host# set policy policy-name
```

You can configure multiple policies for a CoS rule. Each policy identifies the matching conditions for packet source and destination addresses and for packet applications, and the CoS actions to take on those packets. Once a policy in the rule matches a packet, that policy is applied and no other policies in the rule are processed.

4. Specify one or more port-based applications that match the policy.

```
[edit services cos rule rule-name policy policy-name]
user@host# set match application [application-names]
```

5. Specify the destination address that matches the policy.

```
[edit services cos rule rule-name policy policy-name]
user@host# set match destination-address address
```

6. Specify a range of destination addresses that match the policy.

```
[edit services cos rule rule-name policy policy-name]
user@host# set match destination-address-range low minimum-value high maximum-value
```

7. Specify the destination port number that matches the policy.

```
[edit services cos rule rule-name policy policy-name]
user@host# set match destination-port port-number
```

8. Specify the source address that matches the policy.

```
[edit services cos rule rule-name policy policy-name]
user@host# set match source-address address
```

9. Specify a range of source addresses that match the policy.

```
[edit services cos rule rule-name policy policy-name]
user@host# set match source-address-range low minimum-value high maximum-value
```

10. Specify a prefix list of source address prefixes that match the policy.

```
[edit services cos rule rule-name policy policy-name]
user@host# set match source-prefix-list list-name
```

You configure a prefix list by using the `prefix-list` statement at the `[edit policy-options]` hierarchy level.

11. Specify the application profile that defines the CoS policy actions for FTP and SIP traffic.

```
[edit services cos rule rule-name policy policy-name]
user@host# set then application-profile profile-name
```

12. Specify the DSCP value to apply to the packet.

```
[edit services cos rule rule-name policy policy-name]
user@host# set then dscp (alias | bits)
```

The DSCP can be either a code point alias or a DSCP bit value.

13. Specify the forwarding class name to apply to the packet.

```
[edit services cos rule rule-name policy policy-name]
user@host# set then forwarding-class class-name
```

The choices are:

- assured-forwarding
- best-effort
- expedited-forwarding
- network-control
- user-defined classifiers.

You can define classifiers under [edit class-of-service classifiers dscp] hierarchy.

14. Configure system logging for the CoS rule policy.

15. Specify the treatment of flows in the reverse direction of the matching direction. Perform only one of the following:

a. Configure unique values for the reverse direction:

```
[edit services cos rule rule-name policy policy-name]
user@host# set then reverse application-profile profile-name
user@host# set then reverse dscp (alias | bits)
user@host# set then reverse forwarding-class class-name
```

b. Apply the CoS rule policy actions to flows in the reverse direction as well as to flows in the matching direction.

```
[edit services cos rule rule-name policy policy-name]
user@host# set then reflexive
```

c. Store the DSCP and forwarding class of a packet that is received in the match direction of the rule and then apply that DSCP and forwarding class to packets that are received in the reverse direction of the same session.

```
[edit services cos rule rule-name policy policy-name]
user@host# set then revert
```

Configuring Application Profiles for CoS Rules

Configure CoS actions for FTP and SIP traffic. The application profile can then be used in CoS rule actions.

1. Configure a name for the application profile.

```
user@host# edit services cos application-profile profile-name
```

2. Specify the DSCP value to apply to the FTP or SIP (voice or video) packets.

For FTP traffic:

```
[edit services cos application-profile profile-name]
user@host# set ftp data dscp (alias | bits)
```

For SIP voice or video traffic:

```
[edit services cos application-profile profile-name]
user@host# set sip video | voice dscp dscp
```

The DSCP can be either a code point alias or a DSCP bit value.

3. Specify the forwarding class to apply to FTP or SIP packets.

For FTP traffic:

```
[edit services cos application-profile profile-name]
user@host# set ftp data forwarding-class class-name
```

For SIP voice or video traffic:

```
[edit services cos application-profile profile-name]
user@host# set sip video | voice forwarding-class forwarding-class dscp
```

The choices are:

- assured-forwarding
- best-effort
- expedited-forwarding
- network-control

Configuring CoS Rule Sets

A CoS rule set lets you specify a set of services CoS rules. You can then assign the rule set to a service set, which processes the rules in the order they appear. Once a rule matches the packet, the router performs the corresponding action, and no further rules in the rule set are applied.

1. Configure a name for the CoS rule set.

```
user@host# edit services cos rule-set rule-set-name
```

2. Specify the CoS rules that belong to the rule set.

```
[edit services cos rule-set rule-set-name]  
user@host# set rule [rule-name]
```

Configuring the Service Set for CoS

You must apply CoS rules to a service set before the rules can be applied to traffic. Only stateful firewall and NAT rules can be used with CoS rules in a service set.

To configure a service set with CoS rules:

1. Define the service set.

```
[edit services]  
user@host# edit service-set service-set-name
```

2. Configure either an interface service set, which requires a single service interface, or a next-hop service set, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]  
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]  
user@host# set next-hop-service inside-service-interface interface-name outside-service-  
interface interface-name
```

3. Specify the CoS rules to be used with the service set. You can either specify individual rules or rule sets.

To apply individual CoS rules:

```
[edit services service-set service-set-name]  
user@host# set cos-rules [cos-rule-name]
```

To apply CoS rule sets:

```
[edit services service-set service-set-name]  
user@host# set cos-rule-sets [cos-rule-set-name]
```

The service set processes the CoS rules or rule sets in the order in which they appear in the service set configuration.

4. (Optional) Assign at least one stateful firewall rule or NAT rule to the service set.
5. (Optional) Configure the service set to create a CoS session even if a packet is first received in the reverse direction of the matching direction of the CoS rule. The CoS rule values are then applied as soon as a packet in the correct match direction is received.

```
[edit services service-set service-set-name]  
user@host# set cos-options match-rules-on-reverse-flow
```

SEE ALSO

Class of Service Overview for Services PICs (Next Gen Services)

3

PART

Stateful Firewall Services

- [Stateful Firewall Services Overview and Configuration | 324](#)
-

Stateful Firewall Services Overview and Configuration

IN THIS CHAPTER

- [Stateful Firewall Overview for Next Gen Services | 324](#)
- [Configuring Stateful Firewalls for Next Gen Services | 327](#)

Stateful Firewall Overview for Next Gen Services

IN THIS SECTION

- [Benefits | 325](#)
- [Flows and Conversations | 325](#)
- [Stateful Firewall Rules | 325](#)
- [Stateful Firewall Anomaly Checking | 326](#)

Services PICs employ a type of firewall called a stateful firewall. Contrasted with a stateless firewall, which inspects packets in isolation, a stateful firewall provides an extra layer of security by using state information derived from past communications and other applications to make dynamic control decisions for new communication attempts.

Stateful firewalls group relevant flows into conversations, and decide whether the conversation is allowed to be established. If a conversation is allowed, all flows within the conversation are permitted, including flows that are created during the life cycle of the conversation.

Benefits

By inspecting the application protocol data of a flow, the stateful firewall intelligently enforces security policies and permits only the minimally required packet traffic.

Flows and Conversations

A typical Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) conversation consists of two flows: the initiation flow and the responder flow. However, some conversations, such as an FTP conversation, might consist of two control flows and many data flows.

A flow is identified by the following five properties:

- Source address
- Source port
- Destination address
- Destination port
- Protocol

Stateful Firewall Rules

Stateful firewall rules govern whether the conversation is allowed to be established. A rule consists of matching conditions and actions to take.

Matching conditions include direction, source address, destination address, and application protocol or service. In addition to the specific values you configure, you can assign the value *any*, *any-ipv4*, *any-ipv6*, or you can use an *address-book* under *services* to define address lists and ranges for use within stateful firewall rules. Finally, you can specify matches that result in the rule *not* being applied.

Actions in a stateful firewall rule include allowing the traffic or dropping the traffic.

Stateful firewall rules are directional. For each new conversation, the router software determines whether the initiation flow direction matches the rule direction.

Stateful firewall rules are ordered. The software checks the rules in the order in which you include them in the configuration. The first time the software finds a matching rule for a flow, the router implements the action specified by that rule, and ignores subsequent rules.

The stateful firewall rules are configured in relation to an interface. By default, the stateful firewall allows all sessions initiated from the hosts behind the interface to pass through the router.

Stateful Firewall Anomaly Checking

The stateful firewall recognizes the following events as anomalies and sends them to the IDS software for processing:

- IP anomalies:
 - IP version is not correct.
 - IP header length field is too small.
 - IP header length is set larger than the entire packet.
 - Bad header checksum.
 - IP total length field is shorter than header length.
 - Packet has incorrect IP options.
 - Internet Control Message Protocol (ICMP) packet length error.
 - Time-to-live (TTL) equals 0.
- IP address anomalies:
 - IP packet source is broadcast or multicast.
 - Land attack (source IP equals destination IP).
- IP fragmentation anomalies:
 - IP fragment overlap.
 - IP fragment missed.
 - IP fragment length error.
 - IP packet length is more than 64 kilobytes (KB).
 - Tiny fragment attack.
- TCP anomalies:
 - TCP port 0.
 - TCP sequence number 0 and flags 0.
 - TCP sequence number 0 and FIN/PSH/RST flags set.
 - TCP flags with wrong combination (TCP FIN/RST or SYN/(URG|FIN|RST)).

- Bad TCP checksum.
- UDP anomalies:
 - UDP source or destination port 0.
 - UDP header length check failed.
 - Bad UDP checksum.
- Anomalies found through stateful TCP or UDP checks:
 - SYN followed by SYN-ACK packets without ACK from initiator.
 - SYN followed by RST packets.
 - SYN without SYN-ACK.
 - Non-SYN first flow packet.
 - ICMP unreachable errors for SYN packets.
 - ICMP unreachable errors for UDP packets.
- Packets dropped by stateful firewall rules.

Configuring Stateful Firewalls for Next Gen Services

IN THIS SECTION

- [Configuring Stateful Firewall Rules for Next Gen Services | 327](#)
- [Configuring Stateful Firewall Rule Sets for Next Gen Services | 330](#)
- [Configuring the Service Set for Stateful Firewalls for Next Gen Services | 330](#)

To configure stateful firewalls, you configure stateful firewall rules, and apply those rules to a service set. You can also configure stateful firewall rule sets, which contain a set of stateful firewall rules.

Configuring Stateful Firewall Rules for Next Gen Services

A stateful firewall rule specifies which traffic is processed and what action to apply to the traffic.

To configure a stateful firewall rule:

1. Configure a name for the stateful firewall rule.

```
user@host# edit services policies stateful-firewall-rule rule-name
```

2. Specify the traffic flow direction to which the stateful firewall rule applies.

```
[edit services policies stateful-firewall-rule rule-name]
user@host# set match-direction (input | input-output | output)
```

If you configure `input-output`, the rule is applied to sessions initiated from either direction.

If this stateful firewall rule is applied to an interface-type service set, the direction is determined by whether a packet is entering or leaving the interface on which the service set is applied. If this stateful firewall rule is applied to a next-hop service set, the direction is input if the inside interface is used to route the packet, and the direction is output if the outside interface is used to route the package.

3. Configure a name for a policy.

```
[edit services policies stateful-firewall-rule rule-name]
user@host# set policy policy-name
```

You can configure multiple policies for a stateful firewall rule. Each policy identifies the matching conditions for a flow, and whether or not to allow the flow. Once a policy in the rule matches a packet, that policy is applied and no other policies in the rule are processed.

4. Specify the destination address of the flows to which the policy applies.

```
[edit services policies stateful-firewall-rule rule-name policy policy-name]
user@host# set match destination-address (address | any | any-ipv4 | any-ipv6)
```

Alternatively, you can specify an `address-book` under the `services` configuration hierarchy to use in this step.

The destination address can be IPv4 or IPv6.

5. Specify the destination address of the flows to which the policy does not apply.

```
[edit services policies stateful-firewall-rule rule-name policy policy-name]
user@host# set match destination-address-excluded address
```

The destination address can be IPv4 or IPv6.

- Specify the source address of the flows to which the policy applies.

```
[edit services policies stateful-firewall-rule rule-name policy policy-name]
user@host# set match source-address (address | any | any-ipv4 | any-ipv6)
```

Alternatively, you can specify an address-book under the services configuration hierarchy to use in this step.

The source address can be IPv4 or IPv6.

- Specify the source address of the flows to which the policy does not apply.

```
[edit services policies stateful-firewall-rule rule-name policy policy-name]
user@host# set match source-address-excluded address
```

The source address can be IPv4 or IPv6.

- Specify one or more application protocols to which the policy applies.

```
[edit services policies stateful-firewall-rule rule-name policy policy-name]
user@host# set match application [application-name]
```

Use an application protocol definition you have configured at the [edit applications] hierarchy level.

- Specify an action that the policy takes.

```
[edit services policies stateful-firewall-rule rule-name policy policy-name]
user@host# set then (count | deny | reject | permit)
```

where:

count Enables a count, in bytes or kilobytes, of all network traffic the policy allows to pass.

deny Drop the packets.

permit Accept the packets and send them to their destination.

reject Drop the packets. For TCP traffic, send a TCP reset (RST) segment to the source host. For UDP traffic, send an ICMP destination unreachable, port unreachable message (type 3, code 3) to the source host.

Configuring Stateful Firewall Rule Sets for Next Gen Services

A stateful firewall rule set lets you specify a set of stateful firewall rules, which are processed in the order in which they appear in the rule set configuration. Once a stateful firewall rule in the rule set matches a packet, that rule is applied and no other rules in the rule set are processed'.

To configure a stateful firewall rule set:

1. Configure a name for the stateful firewall rule set.

```
user@host# edit services policies stateful-firewall-rule-set rule-set-name
```

2. Specify the stateful firewall rules that belong to the rule set.

```
[edit services policies stateful-firewall-rule-set rule-set-name]
user@host# set stateful-firewall-rule [rule-name]
```

Configuring the Service Set for Stateful Firewalls for Next Gen Services

Stateful firewall rules must be assigned to a service set before they can be applied to traffic.

To configure a service set to apply stateful firewall rules:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service set, which requires a single service interface, or a next-hop service set, which requires an inside and outside service interface.

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface interface-name
```

or

```
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface interface-name outside-service-
interface interface-name
```

3. Specify the stateful firewall rules to be used with the service set. You can specify either individual rules or rule sets but not both.

To apply individual stateful firewall rules:

```
[edit services service-set service-set-name]  
user@host# set stateful-firewall-rules [rule-name]
```

To apply stateful firewall rule sets:

```
[edit services service-set service-set-name]  
user@host# set stateful-firewall-rule-sets [rule-set-name]
```

The service set processes the stateful firewall rules or rule sets in the order in which they appear in the service set configuration.

4

PART

Intrusion Detection Services

- [IDS Screens for Network Attack Protection Overview and Configuration | 333](#)
-

IDS Screens for Network Attack Protection

Overview and Configuration

IN THIS CHAPTER

- [Understanding IDS Screens for Network Attack Protection | 333](#)
- [Configuring Network Attack Protection With IDS Screens for Next Gen Services | 337](#)
- [Configuring the TCP SYN cookie | 347](#)

Understanding IDS Screens for Network Attack Protection

IN THIS SECTION

- [Intrusion Detection Services | 333](#)
- [Benefits | 334](#)
- [Session Limits | 334](#)
- [Suspicious Packet Patterns | 335](#)

Intrusion Detection Services

Intrusion detection services (IDS) screens give you a way to identify and drop traffic that is part of a network attack.

In an IDS screen, you can specify:

- The limits on the number of sessions that originate from individual sources or that terminate at individual destinations
- The types of suspicious packets

You can also choose to log an alarm when an IDS screen identifies a packet, rather than drop the packet.

In addition to IDS screens, you can use firewall filters and policers to stop illegal TCP flags and other bad flag combinations, and to specify general rate limiting (see the *Routing Policies, Firewall Filters, and Traffic Policers User Guide*). IDS screens add a more granular level of filtering.

Use firewall filters and stateful firewall filters to filter out traffic that does not need to be processed by an IDS screen.

Benefits

Provides protection against several types of network attacks.

Session Limits

You can use IDS screens to set session limits for traffic from an individual source or to an individual destination. This protects against network probing and flooding attacks. Traffic that exceeds the session limits is dropped. You can specify session limits either for traffic with a particular IP protocol, such as ICMP, or for traffic in general.

You decide whether the limits apply to individual addresses or to an aggregation of traffic from individual subnets of a particular prefix length. For example, if you aggregate limits for IPv4 subnets with a prefix length of 24, traffic from 192.0.2.2 and 192.0.2.3 is counted against the limits for the 192.0.2.0/24 subnet.

Some common network probing and flooding attacks that session limits protect against include:

ICMP Address Sweep	The attacker sends ICMP request probes (pings) to multiple targets. If a target machine replies, the attacker receives the IP address of the target.
ICMP Flood	The attacker floods a target machine by sending a large number of ICMP packets from one or more source IP addresses. The target machine uses up its resources as it attempts to process those ICMP packets, and then it can no longer process valid traffic.
TCP Port Scan	The attacker sends TCP SYN packets from one source to multiple destination ports of the target machine. If the target replies with a SYN-ACK from one or more destination ports, the attacker learns which ports are open on the target.
TCP SYN Flood	The attacker floods a target machine by sending a large number of TCP SYN packets from one or more source IP addresses. The attacker might use real source IP addresses, which results in a completed TCP connection, or might use fake source IP addresses, resulting in the TCP connection not being completed. The target creates states for all the completed and incomplete TCP connections. The target uses up its resources as it attempts to manage the connection states, and then it can no longer process valid traffic.

UDP Flood The attacker floods a target machine by sending a large number of UDP packets from one or more source IP addresses. The target machine uses up its resources as it attempts to process those UDP packets, and then it can no longer process valid traffic.

Session limits for traffic from a source or to a destination include:

- maximum number of concurrent sessions
- maximum number of packets per second
- maximum number of connections per second

IDS screens also install a dynamic filter on the PFEs of line cards for suspicious activity when the following conditions occur:

- Either the packets per second or the number of connections per second for an individual source or destination address exceeds four times the session limit in the IDS screen. (Dynamic filters are not created from IDS screens that use subnet aggregation.)
- The services card CPU utilization percentage exceeds a configured value (default value is 90 percent).

The dynamic filter drops the suspicious traffic at the PFE, without the traffic being processed by the IDS screen. When the packet or connection rate no longer exceeds four times the limit in the IDS screen, the dynamic filter is removed.

Suspicious Packet Patterns

You can use IDS screens to identify and drop traffic with a suspicious packet pattern. This protects against attackers that craft unusual packets to launch denial-of-service attacks.

Suspicious packet patterns and attacks that you can specify in an IDS screen are:

ICMP fragmentation attack	The attacker sends the target ICMP packets that are IP fragments. These are considered suspicious packets because ICMP packets are usually short. When the target receives these packets, the results can range from processing packets incorrectly to crashing the entire system.
Malformed ICMPv6 packets	Malformed ICMPv6 packets can cause damage to the device and network. Examples of malformed IPv6 packets are packets that are too big (message type 2), that have the next header set to routing (43), or that have a routing header set to hop-by hop.
ICMP large packet attack	The attacker sends the target ICMP frames with an IP length greater than 1024 bytes. These are considered suspicious packets because most ICMP messages are small.

Ping of death attack	The attacker sends the target ICMP ping packets whose IP datagram length (ip_len) exceeds the maximum legal length (65,535 bytes) for IP packets, and the packet is fragmented. When the target attempts to reassemble the IP packets, a buffer overflow might occur, resulting in a system crashing, freezing, and restarting.
Bad option attack	The attacker sends the target packets with incorrectly formatted IPv4 options or IPv6 extension headers. This can cause unpredictable issues, depending on the IP stack implementation of routers and the target.
Fragmented IP packets	IP fragments might contain an attacker's attempt to exploit the vulnerabilities in the packet reassembly code of specific IP stack implementations. When the target receives these packets, the results can range from processing the packets incorrectly to crashing the entire system.
IPv6 extension headers	Attackers can maliciously use extension headers for denial-of-service attacks or to bypass filters.
IPv4 options	Attackers can maliciously use IPv4 options for denial-of-service attacks.
IP teardrop attack	The attacker sends the target fragmented IP packets that overlap. The target machine uses up its resources as it attempts to reassemble the packets, and then it can no longer process valid traffic.
IP unknown protocol attack	The attacker sends the target packets with protocol numbers greater than 137 for IPv4 and 139 for IPv6. An unknown protocol might be malicious.
TCP FIN No ACK attack	The attacker sends the target TCP packets that have the FIN bit set but have the ACK bit unset. This can allow the attacker to identify the operating system of the target or to identify open ports on the target.
Land attack	The attacker sends the target spoofed SYN packets that contain the target's IP address as both the destination and the source IP address. The target uses up its resources as it repeatedly replies to itself. In another variation of the land attack, the SYN packets also contain the same source and destination ports.
TCP SYN ACK ACK attack	The attacker initiates Telnet or FTP connections with the target without completing the connections. The target's session table can fill up, resulting in the device rejecting legitimate connection requests.
TCP SYN FIN attack	The attacker sends the target TCP packets that have both the SYN and the FIN bits set. This can cause unpredictable behavior on the target, depending on its TCP stack implementation.

SYN fragment attack	The attacker sends the target SYN packet fragments. The target caches SYN fragments, waiting for the remaining fragments to arrive so it can reassemble them and complete the connection. A flood of SYN fragments eventually fills the host's memory buffer, preventing valid traffic connections.
TCP no flag attack	The attacker sends the target TCP packets containing no flags. This can cause unpredictable behavior on the target, depending on its TCP stack implementation.
TCP WinNuke attack	The attacker sends a TCP segment with the urgent (URG) flag set and destined for port 139 of a target running Windows. This might cause the target machine to crash.

RELATED DOCUMENTATION

[Configuring Network Attack Protection With IDS Screens for Next Gen Services | 337](#)

Configuring Network Attack Protection With IDS Screens for Next Gen Services

IN THIS SECTION

- [Configuring the IDS Screen Name, Direction, and Alarm Option | 337](#)
- [Configuring Session Limits in the IDS Screen | 338](#)
- [Configuring Suspicious Packet Pattern Detection in the IDS Screen | 343](#)
- [Configuring the Service Set for IDS | 346](#)

Configuring the IDS Screen Name, Direction, and Alarm Option

Configure the IDS screen name, traffic direction, and optional alarm.

1. Specify a name for the IDS screen.

```
[edit services screen]
user@host# set ids-option screen-name
```

2. Specify whether the IDS screen is applied to input traffic, output traffic, or both.

```
[edit services screen ids-option screen-name]
user@host# set match-direction (input | input-output | output)
```

3. If you want the IDS screen to log an alarm when packets exceed the session limit, rather than drop packets, configure alarm-without-drop.

```
[edit services screen ids-option screen-name]
user@host# set alarm-without-drop
```

Configuring Session Limits in the IDS Screen

You can use IDS screens to set session limits for traffic from individual addresses or subnets and to individual addresses or subnets. This protects against network probing and flooding attacks. [Table 36 on page 338](#) shows the session limit options that protect against some common network probing and flooding attacks.

Table 36: IDS Screen Options for Network Attacks Type

Network Attack Type	[edit services screen ids-options <i>screen-name</i> limit-sessions] Options to Set
ICMP Address Sweep	<pre>by-source by-protocol icmp { maximum-sessions <i>number</i>; packet-rate <i>number</i>; session-rate <i>number</i>; }</pre>
ICMP Flood	<pre>by-destination by-protocol icmp { maximum-sessions <i>number</i>; packet-rate <i>number</i>; session-rate <i>number</i>; }</pre>

Table 36: IDS Screen Options for Network Attacks Type *(Continued)*

Network Attack Type	[edit services screen ids-options <i>screen-name</i> limit-sessions] Options to Set
TCP Port Scan	<pre>(by-destination by-source) by-protocol tcp { maximum-sessions <i>number</i>; packet-rate <i>number</i>; }</pre>
TCP SYN Flood	<pre>(by-destination by-source) by-protocol tcp { maximum-sessions <i>number</i>; packet-rate <i>number</i>; session-rate <i>number</i>; }</pre>
UDP Flood	<pre>by-destination by-protocol udp { maximum-sessions <i>number</i>; packet-rate <i>number</i>; session-rate <i>number</i>; }</pre>

To configure the session limits in an IDS screen:

1. If you want to apply session limits to an aggregation of all sessions to individual destination subnets or from individual source subnets rather than individual addresses, configure aggregation.
 - a. To apply session limits to an aggregation of all sessions from within an individual IPv4 subnet, specify the subnet prefix length. The range is from 1 through 32.

```
[edit services screen ids-option screen-name aggregations]
user@host# set source-prefix-mask prefix-value
```

For example, the following statement configures an IPv4 prefix length of 24, and sessions from 192.0.2.2 and 192.0.2.3 are counted as sessions from the 192.0.2.0/24/24 subnet.

```
[edit services screen ids-option screen1 aggregations]
user@host# set source-prefix-mask 24
```


- b. To apply session limits to an aggregation of all sessions from within an individual IPv6 subnet, specify the subnet prefix length. The range is from 1 through 128.

```
[edit services screen ids-option screen-name aggregations]
user@host# set source-prefix-ipv6-mask prefix-value
```

For example, the following statement configures an IPv6 prefix length of 64, and sessions from 2001:db8:1234:72a2::2 and 2001:db8:1234:72a2::3 are counted as sessions from the 2001:db8:1234:72a2::/64 subnet.

```
[edit services screen ids-option screen1 aggregations]
user@host# set source-prefix-ipv6-mask 64
```

- c. To apply session limits to an aggregation of all sessions to an individual IPv4 subnet, specify the subnet prefix length. The range is from 1 through 32.

```
[edit services screen ids-option screen-name aggregations]
user@host# set destination-prefix-mask prefix-value
```

- d. To apply session limits to an aggregation of all sessions to an individual IPv6 subnet, specify the subnet prefix length. The range is from 1 through 128.

```
[edit services screen ids-option screen-name aggregations]
user@host# set destination-prefix-ipv6-mask prefix-value
```

2. If you want to apply session limits from a source for a particular IP protocol:

- a. Configure the maximum number of concurrent sessions allowed from an individual source IP address or subnet for a particular IP protocol.

```
[edit services screen ids-option screen-name limit-session by-source ]
user@host# set by-protocol (icmp | tcp | udp) maximum-sessions number
```

- b. Configure the maximum number of packets per second allowed from an individual source IP address or subnet for a particular protocol.

```
[edit services screen ids-option screen-name limit-session by-source ]
user@host# set by-protocol (icmp | tcp | udp) packet-rate number
```

- c. Configure the maximum number of connections per second allowed from an individual source IP address or subnet for a particular protocol.

```
[edit services screen ids-option screen-name limit-session by-source ]
user@host# set by-protocol (icmp | tcp | udp) session-rate number
```

3. If you want to apply session limits to a destination for a particular IP protocol:

- a. Configure the maximum number of concurrent sessions allowed to an individual destination IP address or subnet for a particular IP protocol.

```
[edit services screen ids-option screen-name limit-session by-destination]
user@host# set by-protocol (icmp | tcp | udp) maximum-sessions number
```

- b. Configure the maximum number of packets per second allowed to an individual destination IP address or subnet for a particular protocol.

```
[edit services screen ids-option screen-name limit-session by-destination ]
user@host# set by-protocol (icmp | tcp | udp) packet-rate number
```

- c. Configure the maximum number of connections per second allowed to an individual destination IP address or subnet for a particular protocol.

```
[edit services screen ids-option screen-name limit-session by-destination ]
user@host# set by-protocol (icmp | tcp | udp) session-rate number
```

4. If you want to apply session limits from a source regardless of the IP protocol:

- a. Configure the maximum number of concurrent sessions allowed from an individual source IP address or subnet.

```
[edit services screen ids-option screen-name limit-session by-source ]
user@host# set maximum-sessions number
```

- b. Configure the maximum number of packets per second allowed from an individual source IP address or subnet

```
[edit services screen ids-option screen-name limit-session by-source ]
user@host# set packets-rate number
```

- c. Configure the maximum number of connections per second allowed from an individual source IP address or subnet.

```
[edit services screen ids-option screen-name limit-session by-source ]
user@host# set session-rate number
```

5. If you want to apply session limits to a destination regardless of the IP protocol:

- a. Configure the maximum number of concurrent sessions allowed to an individual destination IP address or subnet.

```
[edit services screen ids-option screen-name limit-session by-destination ]
user@host# set maximum-sessions number
```

- b. Configure the maximum number of packets per second allowed to an individual destination IP address or subnet

```
[edit services screen ids-option screen-name limit-session by-destination ]
user@host# set packets-rate number
```

- c. Configure the maximum number of connections per second allowed to an individual destination IP address or subnet.

```
[edit services screen ids-option screen-name limit-session by-destination]
user@host# set session-rate number
```

6. Specify the services card CPU utilization percentage that triggers the installation of a dynamic filter on the PFEs of the line cards for suspicious traffic. The default value is 90.

```
[edit services screen]
user@host# set cpu-throttle percentage percent
```

In addition to the CPU utilization percentage threshold, the packet rate or connection rate for an individual source or destination address must exceed four times the session limit in the IDS screen before the dynamic filter is installed. Dynamic filters are not created from IDS screens that use subnet aggregation.

The dynamic filter drops the suspicious traffic at the PFE, without the traffic being processed by the IDS screen. When the packet or connection rate no longer exceeds four times the limit in the IDS screen, the dynamic filter is removed.

Configuring Suspicious Packet Pattern Detection in the IDS Screen

You can use IDS screens to identify and drop suspicious packets. This protects against attackers that craft unusual packets to launch denial-of-service attacks.

To configure suspicious pattern detection:

1. To protect against ICMP fragmentation attacks, identify and drop ICMP packets that are IP fragments.

```
[edit services screen ids-option screen-name icmp]
user@host# set fragment
```

2. To identify and drop malformed ICMPv6 packets, configure `icmpv6-malformed`.

```
[edit services screen ids-option screen-name icmp]
user@host# set icmpv6-malformed
```

3. To protect against ICMP large packet attacks, identify and drop ICMP packets that are larger than 1024 bytes.

```
[edit services screen ids-option screen-name icmp]
user@host# set large
```

4. To protect against ping of death attacks, identify and drop oversized and irregular ICMP packets.

```
[edit services screen ids-option screen-name icmp]
user@host# set ping-death
```

5. To protect against bad option attacks, identify and drop packets with incorrectly formatted IPv4 options or IPv6 extension headers.

```
[edit services screen ids-option screen-name ip]
user@host# set bad-option
```

6. To identify and drop fragmented IP packets, configure `block-frag`.

```
[edit services screen ids-option screen-name ip]
user@host# set block-frag
```

- To drop IPv6 packets with particular extension header values, specify the values.

```
[edit services screen ids-option screen-name ip]
user@host# set ipv6-extension-header header
```

The following header values can be configured:

ah-header	Authentication Header extension header														
esp-header	Encapsulating Security Payload extension header														
fragment-header	Fragment Header extension header														
hop-by-hop-header	Hop-by-Hop option with the specified option: <table> <tr> <td>CALIPSO-option</td><td>Common Architecture Label IPv6 Security Option</td></tr> <tr> <td>jumbo-payload-option</td><td>IPv6 jumbo payload option</td></tr> <tr> <td>quick-start-option</td><td>IPv6 quick start option</td></tr> <tr> <td>router-alert-option</td><td>IPv6 router alert option</td></tr> <tr> <td>RPL-option</td><td>Routing Protocol for Low-Power and Lossy Networks option</td></tr> <tr> <td>SFM-DPD-option</td><td>Simplified Multicast Forwarding IPv6 Duplicate Packet Detection option</td></tr> <tr> <td>user-defined-option-type <i>type-low to type-high</i></td><td> A range of header types <ul style="list-style-type: none"> • Range: 1 through 255. </td></tr> </table>	CALIPSO-option	Common Architecture Label IPv6 Security Option	jumbo-payload-option	IPv6 jumbo payload option	quick-start-option	IPv6 quick start option	router-alert-option	IPv6 router alert option	RPL-option	Routing Protocol for Low-Power and Lossy Networks option	SFM-DPD-option	Simplified Multicast Forwarding IPv6 Duplicate Packet Detection option	user-defined-option-type <i>type-low to type-high</i>	A range of header types <ul style="list-style-type: none"> • Range: 1 through 255.
CALIPSO-option	Common Architecture Label IPv6 Security Option														
jumbo-payload-option	IPv6 jumbo payload option														
quick-start-option	IPv6 quick start option														
router-alert-option	IPv6 router alert option														
RPL-option	Routing Protocol for Low-Power and Lossy Networks option														
SFM-DPD-option	Simplified Multicast Forwarding IPv6 Duplicate Packet Detection option														
user-defined-option-type <i>type-low to type-high</i>	A range of header types <ul style="list-style-type: none"> • Range: 1 through 255. 														
mobility-header	Mobility Header extension header.														
routing-header	Routing Header extension header.														

- To drop IPv4 packets with particular IPv4 option values, specify the values.

```
[edit services screen ids-option screen-name ip]
user@host# set option
```

The following IPv4 option values can be configured:

loose-source-route-option	IP option of 3 (Loose Source Routing)
record-route-option	IP option of 7 (Record Route)
security-option	IP option of 2 (Security)
source-route-option	IP option of 3 (Loose Source Routing) or the IP option of 9 (Strict Source Routing)
stream-option	IP option of 8 (Stream ID)
strict-source-route-option	IP option of 9 (Strict Source Routing)
timestamp-option	IP option of 4 (Internet timestamp)

9. To protect against IP teardrop attacks, identify and drop fragmented IP packets that overlap.

```
[edit services screen ids-option screen-name ip]
user@host# set tear-drop
```

10. To protect against IP unknown protocol attacks, identify and drop IP frames with protocol numbers greater than 137 for IPv4 and 139 for IPv6.

```
[edit services screen ids-option screen-name ip]
user@host# set unknown-protocol
```

11. To protect against TCP FIN No ACK Attacks, identify and drop any packet with the FIN flag set and without the ACK flag set.

```
[edit services screen ids-option screen-name tcp]
user@host# set fin-no-ack
```

12. To protect against land attacks, identify and drop SYN packets that have the same source and destination address or port.

```
[edit services screen ids-option screen-name tcp]
user@host# set land
```

13. To protect against TCP SYN ACK ACK attacks, configure the maximum number of connections from an IP address that can be opened without being completed.

```
[edit services screen ids-option screen-name tcp]
user@host# set syn-ack-ack-proxy number
```

14. To protect against TCP SYN FIN attacks, identify and drop packets that have both the SYN and FIN flags set.

```
[edit services screen ids-option screen-name tcp]
user@host# set syn-fin
```

15. To protect against SYN fragment attacks, identify and drop SYN packet fragments.

```
[edit services screen ids-option screen-name tcp]
user@host# set syn-frag
```

16. To protect against TCP no flag attacks, identify and drop TCP packets that have no flag fields set.

```
[edit services screen ids-option screen-name tcp]
user@host# set tcp-no-flag
```

17. To protect against TCP WinNuke attacks, identify and drop TCP segments that are destined for port 139 and have the urgent (URG) flag set.

```
[edit services screen ids-option screen-name tcp]
user@host# set winnuke
```

Configuring the Service Set for IDS

Configure a service set to apply the IDS screen.

1. Assign the IDS screen to a service set.

```
[edit services]
user@host# set service-set service-set-name ids-option screen-name
```

If the service set is associated with an AMS interface, then the session limits you configure are applicable to each member interface.

2. Limit the packets that the IDS screen processes by configuring a stateful firewall rule . The stateful firewall rule can identify either the traffic that should undergo IDS processing or the traffic that should skip IDS processing:
 - To allow IDS processing on the traffic that matches the stateful firewall rule, include `accept` at the `[edit services stateful-firewall rule rule-name term term-name then]` hierarchy level.
 - To skip IDS processing on the traffic that matches the stateful firewall rule, include `accept skip-ids` at the `[edit services stateful-firewall rule rule-name term term-name then]` hierarchy level.
3. Assign the stateful firewall rule to the service set.

```
[edit services]
user@host# set service-set service-set-name stateful-firewall-rules rule-name
```

4. To protect against header anomaly attacks, configure a header integrity check for the service set.

```
[edit services]
user@host# set service-set service-set-name service-set-options header-integrity-check enable-
all
```

RELATED DOCUMENTATION

[Understanding IDS Screens for Network Attack Protection | 333](#)

Configuring the TCP SYN cookie

IN THIS SECTION

- [Overview | 348](#)
- [Requirements | 348](#)
- [Configuration | 348](#)

Overview

SYN cookie is a stateless SYN proxy mechanism, and you can use it in conjunction with other defenses against a SYN flood attack. This example shows how to configure the TCP SYN cookie.

Requirements

This example uses the following hardware and software components:

- MX480, and MX960 with MX-SPC3
- Junos OS Release 21.2R1

Configuration

IN THIS SECTION

- [Results | 349](#)

To configure the SYN cookie for the TCP protocol for source and/or destination perform these tasks:

1. Set a value for maximum segment size (MSS) to be used for source TCP protocol.

```
[edit]
user@host# set services screen ids-option ids-option-in limit-session by-source by-protocol tcp syn-cookie
mss 64
```

2. Set a value for threshold-rate for source TCP protocol.

```
[edit]
user@host# set services screen ids-option ids-option-in limit-session by-source by-protocol tcp syn-cookie
threshold-rate 100
```

3. Set a value for threshold-num for source TCP protocol

```
[edit]
user@host# set services screen ids-option ids-option-in limit-session by-source by-protocol tcp syn-cookie
threshold-num 100
```

4. Set a value for maximum segment size (MSS) to be used for destination TCP protocol.

```
[edit]
user@host# set services screen ids-option ids-option-in limit-session by-dest by-protocol tcp syn-cookie mss
200
```

5. Set a value for threshold-rate for destination TCP protocol.

```
[edit]
user@host# set services screen ids-option ids-option-in limit-session by-dest by-protocol tcp syn-cookie
threshold-rate 100
```

6. Set a value for threshold-num for destination TCP protocol

```
[edit]
user@host# # set services screen ids-option ids-option-in limit-session by-dest by-protocol tcp syn-cookie
threshold-num 100
```

Results

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

```
user@host# show services screen
```

```
ids-option ids-option-in {
  match-direction input-output;
  limit-session {
    by-source {
      by-protocol {
        tcp {
          syn-cookie {
            mss 64;
            threshold-rate 100;
            threshold-num 100;
          }
        }
      }
    }
  }
  by-destination {
    maximum-sessions 5000;
    session-rate 5000;
    by-protocol {
      tcp {
        syn-cookie {
          mss 200;
          threshold-rate 100;
        }
      }
    }
  }
}
```

```
        threshold-num 100;  
    }
```

5

PART

Traffic Load Balancing

- [Traffic Load Balancing Overview and Configuration | 352](#)
-

Traffic Load Balancing Overview and Configuration

IN THIS CHAPTER

- Traffic Load Balancer Overview | 352
- Configuring TLB | 364

Traffic Load Balancer Overview

IN THIS SECTION

- Traffic Load Balancing Support Summary | 352
- Traffic Load Balancer Application Description | 353
- Traffic Load Balancer Modes of Operation | 354
- Traffic Load Balancer Functions | 357
- Traffic Load Balancer Application Components | 358
- Traffic Load Balancer Configuration Limits | 362

Traffic Load Balancing Support Summary

Table 37 on page 353 provides a summary of the traffic load balancing support on the MS-MPC and MS-MIC cards for Adaptive Services versus support on the MX-SPC3 security services card for Next Gen Services.

Table 37: Traffic Load Balancing Support Summary

	MS-MPC		MX-SPC3
Junos Release	< 16.1R6 & 18.2.R1	≥ 16.1R6 & 18.2R1	19.3R2
Max # of Instances per Chassis	32	2,000 / 32 in L2 DSR mode	2,000
Max # of Virtual Services per Instance	32	32	32
Max # of virtual IP address per virtual service		1	1
Max # of Groups per Instances	32	32	32
Max # of Real-Services (Servers) per Group	255	255	255
Max # of groups per virtual service		1	1
Max # of Network Monitor Profiles per Group		2	2
Max # of HC's per security services per PIC/NPU in 5-sec's		4,000	1,250 – 19.3R2 10,000 – 20.1R1
Supported Health Check Protocols	ICMP, TCP, UDP, HTTP, SSL, Custom		ICMP, TCP, UDP, HTTP, SSL, TLS Hello, Custom

Traffic Load Balancer Application Description

Traffic Load Balancer (TLB) is supported on MX Series routers with either of the Multiservices Modular Port Concentrator (MS-MPC), Multiservices Modular Interface Card (MS-MIC), or the MX Security

Services Processing Card (MX-SPC3) and in conjunction with the Modular Port Concentrator (MPC) line cards supported on the MX Series routers as described in [Table 38 on page 354](#).



NOTE: You cannot run Deterministic NAT and TLB simultaneously.

Table 38: TLB MX Series Router Platform Support Summary

TLB Mode	MX Platform Coverage
Multiservices Modular Port Concentrator (MS-MPC)	MX240, MX2480, MX960, MX2008, MX2010, MX2020
MX Security Services Processing Card (MX-SPC3)	MX240, MX480, MX960

- TLB enables you to distribute traffic among multiple servers.
- TLB employs an MS-MPC-based control plane and a data plane using the MX Series router forwarding engine.
- TLB uses an enhanced version of equal-cost multipath (ECMP). Enhanced ECMP facilitates the distribution of flows across groups of servers. Enhancements to native ECMP ensure that when servers fail, only flows associated with those servers are impacted, minimizing the overall network churn on services and sessions.
- TLB provides application-based health monitoring for up to 255 servers per group, providing Intelligent traffic steering based on health checking of server availability information. You can configure an aggregated multiservices (AMS) interface to provide one-to-one redundancy for MS-MPCs or Next Gen Services MX-SPC3 card used for server health monitoring.
- TLB applies its flow distribution processing to ingress traffic.
- TLB supports multiple virtual routing instances to provide improved support for large scale load balancing requirements.
- TLB supports static virtual-IP-address-to-real-IP-address translation, and static destination port translation during load balancing.

Traffic Load Balancer Modes of Operation

Traffic Load Balancer provides three modes of operation for the distribution of outgoing traffic and for handling the processing of return traffic.

[Table 39 on page 355](#) summarizes the TLB support and which cards it's supported on.

Table 39: TLB Versus Security Service Cards Summary

Security Service Card	MS-MPC	MX-SPC3
Translate	Yes	Yes
Transparent Layer 3 Direct Server Return	Yes	Yes
Transparent Layer 2 Direct Server Return	Yes	Not Supported

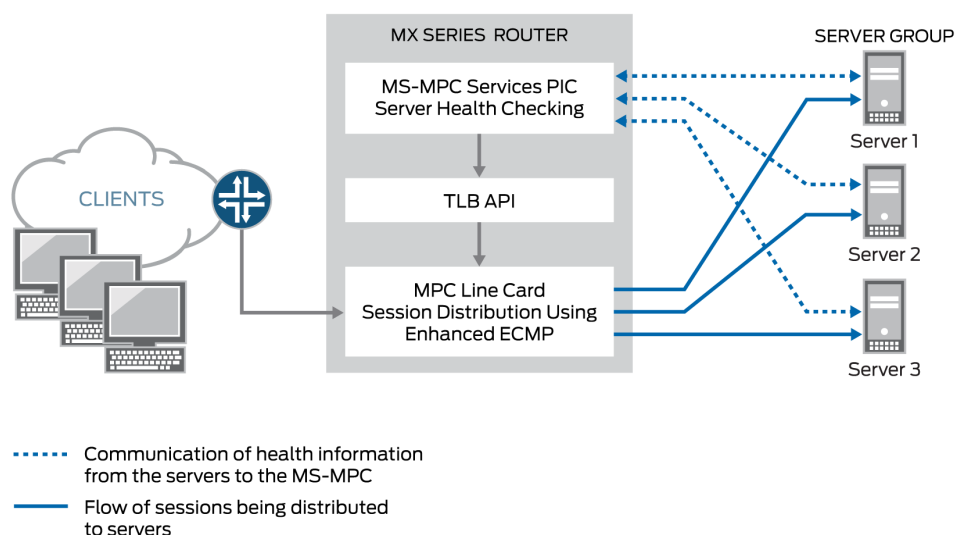
Transparent Mode Layer 2 Direct Server Return

When you use transparent mode Layer 2 direct server return (DSR):

- The PFE processes data.
- Load balancing works by changing the Layer 2 MAC of packets.
- An MS-MPC performs the network-monitoring probes.
- Real servers must be directly (Layer 2) reachable from the MX Series router.
- TLB installs a route and all the traffic over that route is load-balanced.
- TLB never modifies Layer 3 and higher level headers.

[Figure 8 on page 356](#) shows the TLB topology for transparent mode Layer 2 DSR.

Figure 8: TLB Topology for Transparent Mode



Translated Mode

Translated mode provides greater flexibility than transparent mode Layer 2 DSR. When you choose translated mode:

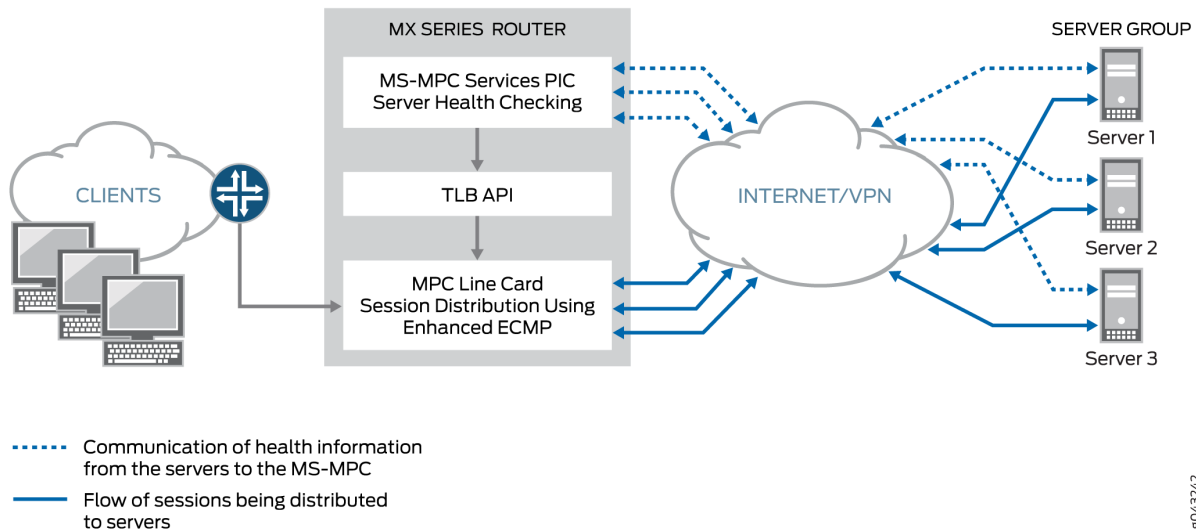
- An MS-MPC performs the network-monitoring probes.
- The PFE performs stateless load balancing:
 - Data traffic directed to a virtual IP address undergoes translation of the virtual IP address to a real server IP address and translates the virtual port to a server listening port. Return traffic undergoes the reverse translation.
 - Client to virtual IP traffic is translated; the traffic is routed to reach its destination.
 - Server-to-client traffic is captured using implicit filters and directed to an appropriate load-balancing next hop for reverse processing. After translation, traffic is routed back to the client.
 - Two load balancing methods are available: random and hash. The random method is only for UDP traffic and provides quavms-random distribution. While not literally random, this mode provides fair distribution of traffic to an available set of servers. The hash method provides a hash key based on any combination of the source IP address, destination IP address, and protocol.



NOTE: Translated mode processing is only available for IPv4-to-IPv4 and IPv6-to-IPv6 traffic.

Figure 9 on page 357 shows the TLB topology for translated mode.

Figure 9: TLB Topology for Translated Mode



Transparent Mode Layer 3 Direct Server Return

Transparent mode Layer 3 DSR load balancing distributes sessions to servers that can be a Layer 3 hop away. Traffic is returned directly to the client from the real-server.

Traffic Load Balancer Functions

TLB provides the following functions:

- TLB always distributes the *requests* for any flow. When you specify DSR mode, the response returns directly to the source. When you specify translated mode, reverse traffic is steered through implicit filters on server-facing interfaces.
- TLB supports hash-based load balancing or random load balancing.
- TLB enables you to configure servers offline to prevent a performance impact that might be caused by a rehashing for all existing flows. You can add a server in the administrative down state and use it later for traffic distribution by disabling the administrative down state. Configuring servers offline helps prevent traffic impact to other servers.
- When health checking determines a server to be down, only the affected flows are rehashed.
- When a previously down server is returned to service, all flows belonging to that server based on hashing return to it, impacting performance for the returned flows. For this reason, you can disable

the automatic rejoining of a server to an active group. You can return servers to service by issuing the `request services traffic-load-balance real-service rejoin operational` command.



NOTE: NAT is not applied to the distributed flows.

- Health check monitoring application runs on an MS-MPC/NPU. This network processor unit (NPU) is not used for handling data traffic.
- TLB supports static virtual-IP-address-to-real-IP-address translation, and static destination port translation during load balancing.
- TLB provides multiple VRF support.

Traffic Load Balancer Application Components

Servers and Server Groups

TLB enables configuration of groups of up to 255 servers (referred to in configuration statements as *real services*) for use as alternate destinations for stateless session distribution. All servers used in server groups must be individually configured before assignment to groups. Load balancing uses hashing or randomization for session distribution. Users can add and delete servers to and from the TLB server distribution table and can also change the administrative status of a server.



NOTE: TLB uses the session distribution next-hop API to update the server distribution table and retrieve statistics. *Applications do not have direct control on the server distribution table management. They can only influence changes indirectly through the add and delete services of the TLB API.*

Server Health Monitoring – Single Health Check and Dual Health Check

TLB supports TCP, HTTP, SSL Hello, TLS Hello, and custom health check probes to monitor the health of servers in a group. You can use a single probe type for a server group, or a dual health check configuration that includes two probe types. The configurable health monitoring function resides on either an MX-SPC3 or an MS-MPC. By default, probe requests are sent every 5 seconds. Also by default, a real server is declared down only after five consecutive probe failures and declared up only after five consecutive probe successes.

Use a custom health check probe to specify the following:

- Expected string in the probe response
- String that is sent with the probe

- Server status to assign when the probe times out (up or down)
- Server status to assign when the expected response to the probe is received (up or down)
- Protocol — UDP or TCP

TLB provides *application stickiness*, meaning that server failures or changes do not affect traffic flows to other active servers. Changing a server's administrative state from up to down does not impact any active flows to remaining servers in the server distribution table. Adding a server or deleting a server from a group has some traffic impact for a length of time that depends on your configuration of the interval and retry parameters in the monitoring profile.

TLB provides two levels of server health monitoring:

- **Single Health Check**—One probe type is attached to a server group by means of the `network-monitoring-profile configuration statement`.
- **TLB Dual Health Check (TLB-DHC)**—Two probe types are associated with a server group by means of the `network-monitoring-profile configuration statement`. A server's status is declared based on the result of two health check probes. Users can configure up to two health check profiles per server group. If a server group is configured for dual health check, a real-service is declared to be UP only when both health-check probes are simultaneously UP; otherwise, a real-service is declared to be DOWN.



NOTE: The following restrictions apply to AMS interfaces used for server health monitoring:

- An AMS interface configured under a TLB instance uses its configured member interfaces exclusively for health checking of configured multiple real servers.
- The member interfaces use unit 0 for single VRF cases, but can use units other than 1 for multiple VRF cases.
- TLB uses the IP address that is configured for AMS member interfaces as the source IP address for health checks.
- The member interfaces must be in the same routing instance as the interface used to reach real servers. This is mandatory for TLB server health-check procedures.

Starting in Junos OS Release 24.2R1, when TLS and SSL are configured in the same group, the OR mechanism is used now instead of AND to determine the status of the real server. That is, the real server is marked as UP if any one of the probes is working. Previously, the real server was marked UP only if both the probes succeeded.

When the SSL probing version is provided, it probes with that version. When the SSL version is not specified, the behavior changes to Fallback from version v3 to v2. The probe starts with SSLv3. If the

SSLv3 probe fails, the system probes for SSLv2. Previously, when the version attribute was not provided explicitly, the probing was done with the default version, v3.



NOTE: This health check behavior enhancement is applicable only when the TLS and SSL probes are configured in the same health check group.

The output for show services traffic-load-balance statistics instance <inst> extensive is changed.

```
user@host# show services traffic-load-balance statistics instance <inst-name>
```

```
Traffic load balance instance name      : <inst-name>
Multi services interface name           : vms-3/0/0
Interface state                         : UP
Interface type                         : Multi services
Route hold timer                       : 180
Active real service count               : 0
Total real service count               : 8
Traffic load balance virtual svc name   : vs1
IP address                             : 60.0.0.1
Virtual service mode                   : Translate mode
Routing instance name                  : fwd_instance_1
Traffic load balance group name         : group1
Traffic load balance group warmup time: 15
Traffic load balance group auto-rejoin: TRUE
Health check interface subunit         : 0
Traffic load balance group down count   : 5
Protocol                               : tcp
Port number                            : 443
Server Listening Port Number            : 443
Route metric                           : 1
Virtual service down count             : 5
Traffic load balance hash method       : source
Network monitoring profile count       : 2
Active real service count               : 0
Total real service count               : 8
Demux Nexthop index                   : 673
Nexthop index                         : 674
Down time                             : 6d 00:01
Total packet sent count                : 361749
Total byte sent count                  : 55165331
Total packet received count            : 542636
Total byte received count              : 28940680
```

```

Network monitoring profile index      : 1
Network monitoring profile name      : nm_prof_ssl
Probe type                          : SSL-HELLO
Probe interval                       : 2
Probe failure retry count            : 5
Probe recovery retry count           : 3
Port number                         : 443
Network monitoring profile index      : 2
Network monitoring profile name      : nm_prof_tls
Probe type                          : TLS-HELLO
Probe interval                       : 5
Probe failure retry count            : 5
Probe recovery retry count           : 5
Port number                         : 443
Traffic load balance real svc name   : rs_1
Routing instance name                : server_vrf_1
IP address                          : 40.1.1.2
Traffic load balance group name      : group1
Admin state                         : UP
Oper state                          : UP
Network monitoring probe up count     : 1
Network monitoring probe down count   : 1
Total rejoin event count              : 8
Total up event count                  : 9
Total down event count                : 9
Real Service packet sent count        : 69804
Real Service byte sent count          : 10644724
Real Service packet received count    : 104706
Real Service byte received count      : 5584336
Total probe sent                      : 358307
Total probe success                   : 76
Total probe fail                      : 358231
Total probe sent failed               : 0
Network monitoring profile index      : 1
Network monitoring profile name      : nm_prof_sslv3
Probe type                          : SSL-HELLO
Probe state                          : UP
SSL probe version                     : 3
Probe sent                           : 255933
Probe success                         : 255879
Probe fail                           : 54
Probe sent failed                     : 0
Probe consecutive success              : 254635

```

```

Probe consecutive fail           : 0
Network monitoring profile index : 2
Network monitoring profile name  : nm_prof_tls
Probe type                      : TLS-HELLO
Probe state                     : DOWN
TLS probe version               : 1.2
Probe sent                     : 102374
Probe success                   : 22
Probe fail                     : 102352
Probe sent failed               : 0
Probe consecutive success       : 0
Probe consecutive fail         : 101854

```



NOTE: The SSL-hello probe version is moved under real server statistics from virtual service when SSL version is not specified under health check profile.

Virtual Services

The virtual service provides a virtual IP address (VIP) that is associated with the group of servers to which traffic is directed as determined by hash-based or random session distribution and server health monitoring. In the case of L2 DSR and L3 DSR, the special address 0.0.0.0 causes all traffic flowing to the forwarding instance to be load balanced.

The virtual service configuration includes:

- Mode—indicating how traffic is handled (translated or transparent).
- The group of servers to which sessions are distributed.
- The load balancing method.
- Routing instance and route metric.



BEST PRACTICE: Although you can assign a virtual address of 0.0.0.0 in order to use default routing, we recommend using a virtual address that can be assigned to a routing instance set up specifically for TLB.

Traffic Load Balancer Configuration Limits

Traffic Load Balancer configuration limits are described in [Table 40 on page 363](#).

Table 40: TLB Configuration Limits

Configuration Component	Configuration Limit
Maximum number of instances	<p>Starting in Junos OS Release 16.1R6 and Junos OS Release 18.2R1, the TLB application supports 2000 TLB instances for virtual services that use the direct-server-return or the translated mode. In earlier releases, the maximum number of instances is 32.</p> <p>If multiple virtual services are using the same server group, then all of those virtual services must use the same load balancing method to support 2000 TLB instances.</p> <p>For virtual services that use the layer2-direct-server-return mode, TLB supports only 32 TLB instances. To perform the same function as the layer2-direct-server-return mode and have support for 2000 TLB instances, you can use the direct-server-return mode and use a service filter with the skip action.</p>
Maximum number of servers per group	255
Maximum number of virtual services per services PIC	32
Maximum number of health checks per services PIC in a 5-second interval	<p>For MS-MPC services cards: 2000</p> <p>For Next Gen Services mode and the MX-SPC3 services cards: 1250</p>
Maximum number of groups per virtual service	1
Maximum number of virtual IP addresses per virtual service	1

Table 40: TLB Configuration Limits *(Continued)*

Configuration Component	Configuration Limit
Supported health checking protocols	<p>ICMP, TCP, HTTP, SSL, TLS-Hello, Custom</p> <p>NOTE: ICMP health checking is supported only on MS-MPC services cards.</p> <p>Starting in Junos OS release 22.4R1, TLB is enhanced to support TLS-Hello health check type. For TLS-Hello over TCP, TLS v1.2 and v1.3 health checks are supported.</p>

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
16.1R6	Starting in Junos OS Release 16.1R6 and Junos OS Release 18.2R1, the TLB application supports 2000 TLB instances for virtual services that use the direct-server-return or the translated mode.

RELATED DOCUMENTATION

- [Interchassis High-Availability](#)
- [Understanding AMS Interfaces](#)

Configuring TLB

IN THIS SECTION

- [Loading the TLB Service Package | 365](#)
- [Configuring a TLB Instance Name | 365](#)
- [Configuring Interface and Routing Information | 366](#)
- [Configuring Servers | 369](#)

- [Configuring Network Monitoring Profiles | 369](#)
- [Configuring Server Groups | 371](#)
- [Configuring Virtual Services | 373](#)
- [Configuring Tracing for the Health Check Monitoring Function | 376](#)

The following topics describe how to configure TLB. To create a complete application, you must also define interfaces and routing information. You can optionally define firewall filters and policy options in order to differentiate TLB traffic.

Loading the TLB Service Package

Load the TLB service package on each service PIC on which you want to run TLB.



NOTE: For Next Gen Services and the MX-SPC3 services card, you do not need to load this package.

To load the TLB service package on a service PIC:

- Load the `jservices-traffic-dird` package.

```
[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]
user@host# set package jservices-traffic-dird
```

For example:

```
[edit chassis fpc 3 pic 0 adaptive-services service-package extension-provider]
user@host# set package jservices-traffic-dird
```

Configuring a TLB Instance Name

Before configuring TLB, enable the `sdk-service` process by configuring system processes `sdk-service enable` at the `[edit]` hierarchy.

To configure a name for the TLB instance:

- At the [edit services traffic-load-balance] hierarchy level, identify the TLB instance name.

```
[edit services traffic-load-balance]
user@host# set instance instance-name
```

For example:

```
[edit services traffic-load-balance]
user@host# set instance tlb-instance1
```

Configuring Interface and Routing Information

To configure interface and routing information:

1. At the [edit services traffic-load-balance instance *instance-name*] hierarchy level, identify the service interface associated with this instance.

```
user@host# [edit services traffic-load-balance instance instance-name]
user@host# set interface interface-name
```

For example, on an MS-MPC:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set interface ms-1/0/0
```

For example, for Next Gen Services on an MX-SPC3:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set interface vms-1/0/0
```

2. Enable the routing of health-check packet responses from real servers to the service interface that you identified in Step 1.

```
[edit interfaces]
user@host# set interface-name unit 0 ip-address-owner service-plane
```

For example, on an MS-MPC:

```
[edit interfaces]
user@host# set ms-1/0/0 unit 0 ip-address-owner service-plane
```

For example, on an MX-SPC3:

```
[edit interfaces]
user@host# set vms-1/0/0 unit 0 ip-address-owner service-plane
```

3. Specify the client interface for which an implicit filter is defined to direct traffic in the forward direction. This is required only for translated mode.

```
user@host# [edit services traffic-load-balance instance instance-name]
user@host# set client-interface interface-name
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set client-interface ge-5/2/0.0
```

4. Specify the virtual routing instance used to route data traffic in the forward direction to servers. This is required for SLT and Layer 3 DSR; it is optional for Layer 2 DSR.

```
user@host# [edit services traffic-load-balance instance instance-name]
user@host# set server-vrf server-vrf
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set server-vrf server-vrf
```

5. Specify the server interface for which implicit filters are defined to direct return traffic to the client.



NOTE: Implicit filters for return traffic are not used for DSR.

```
user@host# [edit services traffic-load-balance instance instance-name]
user@host# set server-interface server-interface
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set server-interface ge-5/2/1.0
```

6. (Optional) Specify the filter used to bypass health checking for return traffic.

```
user@host# [edit services traffic-load-balance instance instance-name]
user@host# set server-inet-bypass-filter server-inet-bypass-filter
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set server-inet-bypass-filter tlb-ipv4-bypass
```

7. Specify the virtual routing instance in which you want the data in the reverse direction to be routed to the clients.

```
user@host# [edit services traffic-load-balance instance instance-name]
user@host# set client-vrf client-vrf
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set client-vrf client-vrf
```



NOTE: Virtual routing instances for routing data in the reverse direction are not used with DSR.

Configuring Servers

To configure servers for the TLB instance:

- Configure a logical name and IP address for each server to be made available for next-hop distribution.

```
[edit services traffic-load-balance instance instance-name]
user@host# set real-service real-service-name address server-ip-address
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1]
user@host# set real-service rs138 address 172.26.99.138
user@host# set real-service rs139 address 172.26.99.139
user@host# set real-service rs140 address 172.26.99.140
```

Configuring Network Monitoring Profiles

A network monitoring profile configures a health check probe, which you assign to a server group to which session traffic is distributed.

To configure a network monitoring profile:

1. Configure the type of probe to use for health monitoring — icmp, tcp, http, ssl-hello, tls-hello, or custom.



NOTE: icmp probes are supported only on MS-MPC cards.

Next Gen Services and the MX-SPC3 do not support ICMP probes in this release.

- For an ICMP probe:

```
[edit services network-monitoring profile profile-name]
user@host.com# set icmp
```

- For a TCP probe:

```
[edit services network-monitoring profile profile-name]
user@host.com# set tcp port tcp-port-number
```

- For an HTTP probe:

```
[edit services network-monitoring profile profile-name]
user@host.com# set http host hostname url url port http-port-number method (get | option)
```

- For an SSL probe:

```
[edit services network-monitoring profile profile-name]
user@host.com# set ssl-hello port port ssl-version
```

- For a TLS-Hello probe:

```
[edit services network-monitoring profile profile-name]
user@host.com# set tls-hello port port number
```

- For a custom probe:

```
[edit services network-monitoring profile profile-name]
user@host.com# set custom cmd priority default-real-service-status (down | up) expect
(ascii | binary) receive-string port port real-service-action (down | up) send (ascii |
binary) send-string
```

2. Configure the interval for probe attempts, in seconds (1 through 180).

```
[edit services network-monitoring profile profile-name]
user@host.com# set probe-interval interval
```

For example:

```
[edit services network-monitoring profile profile1-icmp]
user@host.com# set probe-interval 2
```

3. Configure the number of failure retries, after which the real server is tagged as down.

```
[edit services network-monitoring profile profile-name]
user@host.com# set failure-retries number-of-retries
```

For example:

```
[edit services network-monitoring profile profile1-icmp]
user@host.com# set failure-retries 3
```

4. Configure the number of recovery retries, which is the number of successful probe attempts after which the server is declared up.

```
[edit services network-monitoring profile profile-name]
user@host.com# set recovery-retries number-of-retries
```

For example:

```
[edit services network-monitoring profile profile1-icmp]
user@host.com# set recovery-retries 1
```

Configuring Server Groups

Server groups consist of servers to which traffic is distributed by means of stateless, hash-based session distribution and server health monitoring.

To configure a server group:

1. Specify the names of one or more configured real servers.

```
[edit services traffic-load-balance instance instance-name groups group-name]
user@host.com# set real-services real-service-name, ...
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 groups tlb-group1]
user@host.com# set real-services [ rs138 rs139 rs140 ]
```

2. Configure the routing instance for the group when you do not want to use the default instance, `inet.0`.

```
[edit services traffic-load-balance instance instance-name groups group-name]
user@host.com# set routing-instance routing-instance-name
```


For example:

```
[edit services traffic-load-balance instance tlb-instance1 groups tlb-group1]
user@host.com# set routing-instance tlb-routing-instance1
```

3. (Optional) Disable the default option that allows a server to rejoin the group automatically when it comes up.

```
[edit services traffic-load-balance instance instance-name group group-name]
user@host.com# set real-service-rejoin-options no-auto-rejoin
```

4. (Optional) Configure the logical unit of the instance's service interface to use for health checking.
 - a. Specify the logical unit.

```
[edit services traffic-load-balance instance instance-name group group-name]
user@host.com# set health-check-interface-subunit health-check-interface-subunit
```

- b. Enable the routing of health-check packet responses from real servers to the interface.

```
[edit interfaces]
user@host.com# set interface-name unit subunit ip-address-owner service-plane
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 group tlb-group1]
user@host.com# set health-check-interface-subunit 30
[edit interfaces]
user@host.com# set ms-1/0/0 unit 30 ip-address-owner service-plane
```

5. Configure one or two network monitoring profiles to be used to monitor the health of servers in this group.

```
[edit services traffic-load-balance instance instance-name groups group-name]
user@host.com# set network-monitoring-profile profile-name1 profile-name2
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 groups tlb-group1]
user@host.com# set network-monitoring-profile profile1-icmp profile2-http
```

Configuring Virtual Services

A virtual service provides an address that is associated with a the group of servers to which traffic is directed as determined by hash-based or random session distribution and server health monitoring. You may optionally specify filters and routing instances to steer traffic for TLB.

To configure a virtual service:

1. At the `[edit services traffic-load-balance instance instance-name]` hierarchy level, specify a non-zero address for the virtual service.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-name]
user@host# set address virtual-ip-address
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set address 192.0.2.11
```

2. Specify the server group used for this virtual service.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-name]
user@host# set group group-name
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set group tlb-group1
```

3. (Optional) Specify a routing instance for the virtual service. If you do not specify a routing instance, the default routing instance is used.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-name]
user@host# set routing-instance routing-instance
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set routing-instance msp-tproxy-server-vrf31
```

4. Specify the processing mode for the virtual service.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-name]
user@host# set mode (layer2-direct-server-return | direct-server-return | translated)
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set mode translated
```

5. (Optional) For a translated mode virtual service, enable the addition of the IP addresses for all the real servers in the group under the virtual service to the server-side filters. Doing this allows you to configure two virtual services with the same listening port and protocol on the same interface and VRF.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-name]
user@host# set include-real-server-ips-in-server-filter
```

6. (Optional) Specify a routing metric for the virtual service.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-name]
user@host# set routing-metric routing-metric
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set routing-metric 128
```

7. Specify the method used for load balancing. You can specify a hash method that provides a hash key based on any combination of the source IP address, destination IP address, and protocol, or you can specify `random`.

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set load-balancing-method (hash hash-key (source-ip | destination-ip | proto) |
random)
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set load-balancing-method hash hash-key source-ip
```

or

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set load-balancing-method random
```



NOTE: If you switch between the hash method and the random method for a virtual service, the statistics for the virtual service are lost.

8. For a translated mode virtual service, specify a service for translation, including a virtual-port, server-listening-port, and protocol.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-name]
user@host# set service service-name virtual-port virtual-port server-listening-port server-listening-port protocol (udp | tcp)
```

For example:

```
[edit services traffic-load-balance instance tlb-instance1 virtual-service virtual-service1]
user@host# set service fast-track-service virtual-port 1111 server-listening-port 22 protocol
tcp
```

9. Commit the configuration.

```
[edit services traffic-load-balance instance instance-name virtual-service virtual-service-
name]
user@host# commit
```



NOTE: In the absence of a client-interface configuration under the TLB instance, the implicit client filter (for VIP) is attached to the client-vrf configured under the TLB instance. In this case, the routing-instance under a translate mode virtual service cannot be the same as the client-vrf configured under the TLB instance. If it is, the commit fails.

Configuring Tracing for the Health Check Monitoring Function

To configure tracing options for the health check monitoring function:

1. Specify that you want to configure tracing options for the health check monitoring function.

```
[edit services network-monitoring]
user@host# edit traceoptions
```

2. (Optional) Configure the name of the file used for the trace output.

```
[edit services network-monitoring traceoptions]
user@host# set file file-name
```

3. (Optional) Disable remote tracing capabilities.

```
[edit services network-monitoring traceoptions]
user@host# set no-remote-trace
```

4. (Optional) Configure flags to filter the operations to be logged.

```
[edit services network-monitoring traceoptions]
user@host# set flag flag
```

[Table 41 on page 377](#) describes the flags that you can include.

Table 41: Trace Flags

Flag	Support on MS-MPC and MX-SPC3 Cards	Description
all	MS-MPC and MX-SPC3	Trace all operations.
all-real-services	MX-SPC3	Trace all real services.
config	MS-MPC and MX-SPC3	Trace traffic load balancer configuration events.
connect	MS-MPC and MX-SPC3	Trace traffic load balancer ipc events.
database	MS-MPC and MX-SPC3	Trace database events.
file-descriptor-queue	MS-MPC	Trace file descriptor queue events.
inter-thread	MS-MPC	Trace inter-thread communication events.
filter	MS-MPC and MX-SPC3	Trace traffic load balancer filter programming events.
health	MS-MPC and MX-SPC3	Trace traffic load balancer health events.
messages	MS-MPC and MX-SPC3	Trace normal events.
normal	MS-MPC and MX-SPC3	Trace normal events.
operational-commands	MS-MPC and MX-SPC3	Trace traffic load balancer show events.

Table 41: Trace Flags (*Continued*)

Flag	Support on MS-MPC and MX-SPC3 Cards	Description
parse	MS-MPC and MX-SPC3	Trace traffic load balancer parse events.
probe	MS-MPC and MX-SPC3	Trace probe events.
probe-infra	MS-MPC and MX-SPC3	Trace probe infra events.
route	MS-MPC and MX-SPC3	Trace traffic load balancer route events.
snmp	MS-MPC and MX-SPC3	Trace traffic load balancer SNMP events.
statistics	MS-MPC and MX-SPC3	Trace traffic load balancer statistics events.
system	MS-MPC and MX-SPC3	Trace traffic load balancer system events.

5. (Optional) Configure the level of tracing.

```
[edit services network-monitoring traceoptions]
user@host# set level (all | error | info | notice | verbose | warning)
```

6. (Optional) Configure tracing for a particular real server within a particular server group.

```
[edit services network-monitoring traceoptions]
user@host# set monitor monitor-object-name group-name group-name real-services-name real-  
service-name
```

7. (Optional) Starting in Junos OS Release 16.1R6 and 18.2R1, configure tracing for a particular virtual service and instance.

```
[edit services traffic-load-balance traceoptions]
user@host# set monitor monitor-object-name instance-name instance-name virtual-svc-name virtual-service-name
```

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
16.1R6	Starting in Junos OS Release 16.1R6 and 18.2R1, configure tracing for a particular virtual service and instance.

6

PART

DNS Request Filtering

- [DNS Request Filtering Overview and Configuration | 381](#)
-

DNS Request Filtering Overview and Configuration

IN THIS CHAPTER

- [DNS Request Filtering for Disallowed Website Domains | 381](#)
- [DNS Request Filtering System Logging Error Messages | 403](#)

DNS Request Filtering for Disallowed Website Domains

IN THIS SECTION

- [Overview of DNS Request Filtering | 381](#)
- [How to Configure DNS Request Filtering | 384](#)
- [Multitenant Support for DNS Filtering | 391](#)
- [Configuring Multi-tenant Support for DNS Filtering | 392](#)
- [Example: Configuring Multitenant Support for DNS Filtering | 397](#)

Overview of DNS Request Filtering

IN THIS SECTION

- [Benefits | 383](#)
- [Disallowed Domain Filter Database File | 383](#)
- [DNS Filter Profile | 384](#)

Starting in Junos OS Release 18.3R1, you can configure DNS filtering to identify DNS requests for disallowed website domains. Starting in Junos OS Release 19.3R2, you can configure DNS filtering if you are running Next Gen Services with the MX-SPC3 services card. Next Gen Services are supported on MX240, MX480 and MX960 routers. For DNS request types A, AAAA, MX, CNAME, TXT, SRV, and ANY, you configure the action to take for a DNS request for a disallowed domain. You can either:

- Block access to the website by sending a DNS response that contains the IP address or fully qualified domain name (FQDN) of a DNS sinkhole server. This ensures that when the client attempts to send traffic to the disallowed domain, the traffic instead goes to the sinkhole server (see [Figure 10 on page 383](#)).
- Log the request and allow access.

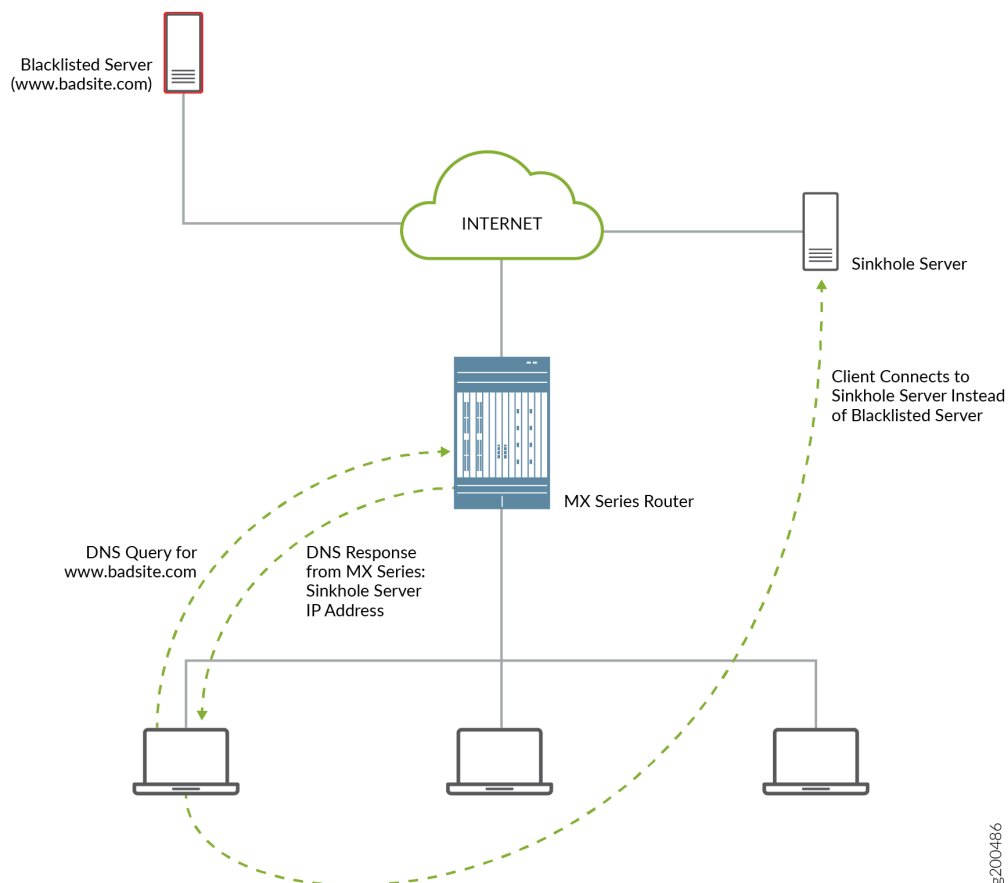
Starting in Junos OS release 21.1R1, you can also configure the following actions for a DNS request for a disallowed domain:

- Alert
- Accept
- Drop
- Drop-no-log

For other DNS request types for a disallowed domain, the request is logged and access is allowed.

The actions that the sinkhole server takes are not controlled by the DNS request filtering feature; you are responsible for configuring the sinkhole server actions. For example, the sinkhole server could send a message to the requestor that the domain is not reachable and prevent access to the disallowed domain.

Figure 10: DNS Request for Disallowed Domain



Benefits

DNS filtering redirects DNS requests for disallowed website domains to sinkhole servers, while preventing anyone operating the system from seeing the list of disallowed domains. This is because the disallowed domain names are in an encrypted format.

Disallowed Domain Filter Database File

DNS request filtering requires a disallowed domain filter database .txt file, which identifies each disallowed domain name, the action to take on a DNS request for the disallowed domain, and the IP address or fully qualified domain name (FQDN) of a DNS sinkhole server.

DNS Filter Profile

You configure a DNS filter profile to specify which disallowed domain filter database file to use. You can also specify the interfaces on which DNS request filtering is performed, limit the filtering to requests for specific DNS servers, and limit the filtering to requests from specific source IP address prefixes.

How to Configure DNS Request Filtering

IN THIS SECTION

- [How to Configure a Domain Filter Database | 384](#)
- [How to Configure a DNS Filter Profile | 385](#)
- [How to Configure a Service Set for DNS Filtering | 391](#)

To filter DNS requests for disallowed website domains, perform the following:

How to Configure a Domain Filter Database

Create one or more domain filter database files that include an entry for each disallowed domain. Each entry specifies what to do with a DNS request for a disallowed website domain.

To configure a domain filter database file:

1. Create the name for the file. The database file name can have a maximum length of 64 characters and must have a **.txt** extension.
2. Add a file header with a format such as
20170314_01:domain,sinkhole_ip,v6_sinkhole,sinkhole_fqdn,id,action.
3. Add an entry in the file for each disallowed domain. You can include a maximum of 10,000 domain entries. Each entry in the database file has the following items:

hashed-domain-name,IPv4 sinkhole address, IPv6 sinkhole address, sinkhole FQDN, ID, action

where:

- **hashed-domain-name** is a hashed value of the disallowed domain name (64 hexadecimal characters). The hash method and hash key that you use to produce the hashed domain value are needed when you configure DNS filtering with the Junos OS CLI.
- **IPv4 sinkhole address** is the address of the DNS sinkhole server for IPv4 DNS requests.
- **IPv6 sinkhole address** is the address of the DNS sinkhole server for IPv6 DNS requests.

- **sinkhole FQDN** is the fully qualified domain name of the DNS sinkhole server.
- **ID** is a 32-bit number that uniquely associates the entry with the hashed domain name.
- **action** is the action to apply to a DNS request that matches the disallowed domain name. If you enter :
 - **replace**, the MX Series router sends the client a DNS response with the IP address or FQDN of the DNS sinkhole server. If you enter **report**, the DNS request is logged and then sent to the DNS server.
 - **report**, the DNS request is logged and then sent to the DNS server.
 - **alert**, the DNS request is logged and the request is sent to the DNS server.
 - **accept**, the DNS request is logged and the request is sent to the DNS server.
 - **drop**, the DNS request is dropped and the request is logged .DNS request is not sent to the DNS server.
 - **drop-no-log**, the DNS request is dropped and no syslog is generated. DNS request is not sent to the DNS server.
- 4. In the last line of the file, include the file hash, which you calculate by using the same key and hash method that you used to produce the hashed domain names.
- 5. Save the database files on the Routing Engine in the **/var/db/url-filterd** directory.
- 6. Validate the domain filter database file.

```
user@host> request services web-filter validate dns-filter-file-name filename hash-key key-string hash-method hash-method-name
```

7. If you make any changes to the database file, apply the changes.

```
user@host> request services web-filter update dns-filter-database filename
```

How to Configure a DNS Filter Profile

A DNS filter profile includes general settings for filtering DNS requests for disallowed website domains, and includes up to 32 templates. The template settings apply to DNS requests on specific uplink and downlink logical interfaces or routing instances, or to DNS requests from specific source IP address prefixes, and override the corresponding settings at the DNS profile level. You can configure up to eight DNS filter profiles.

To configure a DNS filter profile:

1. Configure the name for a DNS filter profile:

```
[edit]
user@host# edit services web-filter profile profile-name
```

The maximum number of profiles is 8.

2. Configure the interval for logging per-client statistics for DNS filtering. The range is 0 through 60 minutes and the default is 5 minutes.

```
[edit services web-filter profile profile-name]
user@host# set global-dns-stats-log-timer minutes
```

3. Configure general DNS filtering settings for the profile. These values are used if a DNS request does not match a specific template.

- a. Specify the name of the domain filter database to use when filtering DNS requests.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set database-file filename
```

- b. (Optional) To limit DNS filtering to DNS requests that are destined for specific DNS servers, specify up to three IP addresses (IPv4 or IPv6).

```
[edit services web-filter profile profile-name dns-filter]
user@host# set dns-server [ ip-address ]
```

- c. Specify the format for the hash key.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set hash-key ascii-text
```

- d. Specify the hash key that you used to create the hashed domain name in the domain filter database file.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set hash-key key-string
```

- e. Specify the hash method that was used to create the hashed domain name in the domain filter database file.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set hash-method hash-method-name
```

The only supported hash method is hmac-sha2-256.

- f. Configure the interval for logging statistics for DNS requests and for sinkhole actions performed for each customer IP address. The range is 1 through 60 minutes and the default is 5 minutes.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set statistics-log-timer minutes
```

- g. Configure the time to live while sending the DNS response after taking the DNS sinkhole action. The range is 0 through 86,400 seconds and the default is 1800.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set dns-resp-ttl seconds
```

- h. Configure the level of subdomains that are searched for a match. The range is 0 through 10. A value of 0 indicates that subdomains are not searched.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set wildcarding-level level
```

For example, if you set the wildcarding-level to 4 and the database file includes an entry for **example.com**, the following comparisons are made for a DNS request that arrives with the domain **198.51.100.0.example.com**:

- **198.51.100.0.example.com**: no match
- **51.100.0.example.com**: no match for one level down
- **100.0.example.com**: no match for two levels down
- **0.example.com**: no match for three levels down
- **example.com**: match for four levels down

4. Configure a template. You can configure a maximum of 8 templates in a profile. Each template identifies filter settings for DNS requests on specific uplink and downlink logical interfaces or routing instances, or for DNS requests from specific source IP address prefixes.

- a. Configure the name for the template.

```
[edit services web-filter profile profile-name]
user@host# set dns-filter-template template-name
```

- b. (Optional) Specify the client-facing logical interfaces (uplink) to which the DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set client-interfaces client-interface-name
```

- c. (Optional) Specify the server-facing logical interfaces (downlink) to which the DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set server-interfaces server-interface-name
```

- d. (Optional) Specify the routing instance for the client-facing logical interface to which the DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set client-routing-instance client-routing-instance-name
```

- e. (Optional) Specify the routing instance for the server-facing logical interface to which DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set server-routing-instance server-routing-instance-name
```



NOTE: If you configure the client and server interfaces or the client and server routing instances, implicit filters are installed on the interfaces or routing instances to direct DNS traffic to the services PIC for DNS filtering. If you configure neither the client and server interfaces nor the routing instances, you must provide a way to direct DNS traffic to the services PIC (for example, via routes).

- f. Specify the name of the domain filter database to use when filtering DNS requests.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set database-file filename
```

- g. (Optional) To limit DNS filtering to DNS requests that are destined for specific DNS servers, specify up to three IP addresses (IPv4 or IPv6).

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set dns-server ip-address
```

- h. Specify the hash method that was used to create the hashed domain name in the domain filter database file.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set hash-method hash-method-name
```

The only supported hash method is hmac-sha2-256.

- i. Specify the hash key that was used to create the hashed domain name in the domain filter database file.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set hash-key key-string
```

- j. Configure the interval for logging statistics for DNS requests and for sinkhole actions performed for each customer IP address. The range is 1 through 60 minutes and the default is 5 minutes.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set statistics-log-timer minutes
```

- k. Configure the time to live while sending the DNS response after taking the DNS sinkhole action. The range is 0 through 86,400 seconds and the default is 1800.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set dns-resp-ttl seconds
```

- l. Configure the level of subdomains that are searched for a match. The range is 0 through 10. A value of 0 indicates that subdomains are not searched.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set wildcarding-level level
```

For example, if you set the `wildcarding-level` to 4 and the database file includes an entry for **example.com**, the following comparisons are made for a DNS request that arrives with the domain **198.51.100.0.example.com**:

- **198.51.100.0.example.com**: no match
- **51.100.0.example.com**: no match for one level down
- **100.0.example.com**: no match for two levels down
- **0.example.com**: no match for three levels down
- **example.com**: match for four levels down

- m. (Optional) Specify the response error code for SRV and TXT query types.
(Optional) Specify the response error code for SRV and TXT query types.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set txt-resp-err-code (Noerror | Refused)
user@host# set srv-resp-err-code (Noerror | Refused)
```

- n. Configure a term for the template. You can configure a maximum of 64 terms in a template.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set term term-name
```

- o. (Optional) Specify the source IP address prefixes of DNS requests you want to filter. You can configure a maximum of 64 prefixes in a term.

```
[edit services web-filter profile profile-name dns-filter-template template-name term term-name]
user@host# set from src-ip-prefix source-prefix
```

- p. Specify that the sinkhole action identified in the domain filter database is performed on disallowed DNS requests.

```
[edit services web-filter profile profile-name dns-filter-template template-name term term-name]
user@host# set then dns-sinkhole
```

How to Configure a Service Set for DNS Filtering

- Associate the DNS filter profile with a next-hop service set and enable logging for DNS filtering. The service interface can be an ms- or vms- interface Next Gen Services with MX-SPC3 services card), or it can be an aggregated multiservices (AMS) interface.

```
[edit services service-set service-set-name]
user@host# set web-filter-profile profile-name
user@host# set syslog host hostname class urlf-logs
user@host# set next-hop-service inside-service-interface interface-name.unit-number
user@host# set next-hop-service outside-service-interface interface-name.unit-number
```

Multitenant Support for DNS Filtering

IN THIS SECTION

- [Overview | 391](#)

Overview

Starting in Junos OS Release 21.1R1, you can configure custom domain feeds per customer or IP subgroup. You can :

- Configure domain names and actions for multiple tenants such that domain feeds can be managed on a per tenant basis.
- Configure hierarchical domain feed management per profile, per dns-filter-template or per dns-filter-term.
- Exempt domain feeds at the IP, subnet, or CIDR level.

To implement the multitenant support for DNS filtering, creating the domain filter database file under template or profile level is disabled. You need not specify a file at the template or profile level. Starting in Junos OS 21.1R1, by default, a global file with a fixed name, **nsf_multi_tenant_dn_custom_file.txt** (plain text format) or **dnsf_multi_tenant_dn_custom_file_hashed.txt** (encrypted file) is available.

Each entry in the database file has the following items:

hashed-domain-name, IPv4 sinkhole address, IPv6 sinkhole address, sinkhole FQDN, ID, action, feed-name.

The file hash is calculated and appended to the list of domain name entries in the file. The file hash is calculated using a global key and method, which is validated with the file hash computed using the hash key configured at the [edit services web-filter] hierarchy. The file validation is successful only if the calculated file-hash matches the file hash present in the file.

Each entry in **nsf_multi_tenant_dn_custom_file.txt** file consists of an additional field called **feed-name**. This **feed-name** is used as an indicator to group set of domain-names and map them to a tenant (profile, template, term, or IP address).

When the DNS packets are received from a particular SRC IP address, the corresponding feed-name is fetched and lookup happens against the domain-names mapped with the feed-name associated with the term. If the feed-name is not provisioned for that IP address, then it falls back to the feed-name configured at the template-level and lookup happens against the domain-names mapped with the feed-name associated with the template. If the feed-name is not configured at template, then the lookup is against the domain-names mapped against the feed-name associated with the profile.

Configuring Multi-tenant Support for DNS Filtering

1. Configure the web filter.

```
[edit]
user@host# edit services web-filter
```

2. Enable multi-tenant support

```
[edit services web-filter]
user@host# set multi-tenant-support
```

3. Configure the global file hash key and hash method.

```
[edit services web-filter]
user@host# set multi-tenant-hash
user@host# set multi-tenant-hash file-hash-key (ascii-text | hexadecimal)
user@host# set multi-tenant-hash hash-method (ascii-text | hexadecimal)
```



NOTE: When `multi-tenant-hash` is configured, it indicates that the global dns feed file consists of only encrypted feeds. When `multi-tenant-hash` is not configured it indicates that the global dns feed file has feeds in plain text format.

4. Configure the name for a DNS filter profile and map the domain feed at the profile level. The feed name indicator configured at the profile level is applied to all the templates and terms under the profile that do not have the feed name indicator configured.

```
[edit]
user@host# [edit services web-filter profile profile-name]
user@host# [edit services web-filter profile profile-name feed-name feed-name]
```

5. Configure general DNS filtering settings for the profile. These values are used if a DNS request does not match a specific template.

- a. (Optional) To limit DNS filtering to DNS requests that are destined for specific DNS servers, specify up to three IP addresses (IPv4 or IPv6).

```
[edit services web-filter profile profile-name dns-filter]
user@host# set dns-server [ip-address]
```

- b. Configure the interval for logging statistics for DNS requests and for sinkhole actions performed for each customer IP address. The range is 1 through 60 minutes and the default is 5 minutes.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set statistics-log-timer minutes
```

- c. Configure the time to live (TTL) to send the DNS response after taking the DNS sinkhole action. The range is 0 through 86,400 seconds and the default is 1800.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set dns-resp-ttl seconds
```

- d. Configure the level of subdomains that are searched for a match. The range is 0 through 10. A value of 0 indicates that subdomains are not searched.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set wildcarding-level level
```

- e. (Optional) Specify the response error code for the TXT query type.

```
[edit services web-filter profile profile-name dns-filter]
user@host# set txt-resp-err-code (Noerror | Refused) level
```

6. Configure a template. You can configure a maximum of 8 templates in a profile. Each template identifies filter settings for DNS requests on specific uplink and downlink logical interfaces or routing instances, or for DNS requests from specific source IP address prefixes.

- a. Configure the name for the template.

```
[edit services web-filter profile profile-name]
user@host# set dns-filter-template template-name
```

- b. Configure the feed name. With multitenant format, you can no longer add a file name under profile or template. The feed name specified under profile has lesser precedence compared to the one configured under the template.

```
[edit services web-filter profile profile-name dns-filter-template template-name ]
user@host# set feed-name feed-name
```

- c. (Optional) Specify the client-facing logical interfaces (uplink) to which the DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set client-interfaces client-interface-name
```

- d. (Optional) Specify the server-facing logical interfaces (downlink) to which the DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set server-interfaces server-interface-name
```

- e. (Optional) Specify the routing instance for the client-facing logical interface to which the DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set client-routing-instance client-routing-instance-name
```

- f. (Optional) Specify the routing instance for the server-facing logical interface to which DNS filtering is applied.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set server-routing-instance server-routing-instance-name
```



NOTE: If you configure the client and server interfaces or the client and server routing instances, implicit filters are installed on the interfaces or routing instances to direct DNS traffic to the services PIC for DNS filtering. If you configure neither the client and server interfaces nor the routing instances, you must provide a way to direct DNS traffic to the services PIC (for example, through routes).

- g. Configure the interval for logging statistics for DNS requests and for sinkhole actions performed for each customer IP address. The range is 1 through 60 minutes and the default is 5 minutes.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set statistics-log-timer minutes
```


- h. Configure the time to live while sending the DNS response after taking the DNS sinkhole action. The range is 0 through 86,400 seconds and the default is 1800.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set dns-resp-ttl seconds
```

- i. Configure the level of subdomains that are searched for a match. The range is 0 through 10. A value of 0 indicates that subdomains are not searched.

```
[edit services web-filter profile profile-name dns-filter-template template-name dns-
filter]
user@host# set wildcarding-level level
```

- j. Configure a term for the template. You can configure a maximum of 64 terms in a template.

```
[edit services web-filter profile profile-name dns-filter-template template-name]
user@host# set term term-name
```

- k. Configure the feed name. The feed name configured at the term takes higher precedence over the one configured under the template. However, if the sinkhole domain is matching the only domain mentioned in the feed name under template, the action specified for that entry is implemented.

```
[edit services web-filter profile profile-name dns-filter-template template-name term term-
name]
user@host# set feed-name feed-name
```

- l. (Optional) Specify the source IP address prefixes of DNS requests you want to filter. You can configure a maximum of 64 prefixes in a term.

```
[edit services web-filter profile profile-name dns-filter-template template-name term term-
name]
user@host# set from src-ip-prefix source-prefix
```

- m. Configure that the sinkhole action identified in the domain filter database is performed on disallowed DNS requests.

```
[edit services web-filter profile profile-name dns-filter-template template-name term term-name]
user@host# set then dns-sinkhole
```

7. Associate the DNS filter profile with a next-hop service set and enable logging for DNS filtering. The service interface can be a multiservices (ms) or virtual multi service (vms) interface (Next Gen Services with MX-SPC3 services card), or it can be an aggregated multiservices (AMS) interface.

```
[edit services service-set service-set-name]
user@host# set syslog mode event
user@host# set syslog syslog event-rate event-rate
user@host# set syslog local-category urlf
user@host# set web-filter-profile profile-name
user@host# set set next-hop-service inside-service-interface interface-name.unit-number
user@host# set set next-hop-service outside-service-interface interface-name.unit-number
```

8. If you are running Next Gen Services on the MX-SPC3 services card, configure the vms interface to get the FPC and PIC information in the syslog.

```
[edit interfaces interface-name]
user@host# set vms 0/0/0
user@host# set services-options
```

```
[edit interfaces interface-name]
user@host# fpc-pic-information
```

Example: Configuring Multitenant Support for DNS Filtering

IN THIS SECTION

- [Configuration](#) | 398

Configuration

IN THIS SECTION

- [CLI Quick Configuration | 398](#)

CLI Quick Configuration

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

```
set services service-set Test Zone3 syslog mode stream
set services service-set Test Zone3 syslog source-address 10.1.1.1
set services service-set Test Zone3 syslog stream t1 category urlf
set services service-set Test Zone3 syslog stream t1 host 10.10.1.1
set services service-set Test Zone3 syslog stream t1 routing-instance client_vr4
set services service-set Test Zone3 web-filter-profile Test-Profile-3-Zone3
set services service-set Test Zone3 next-hop-service inside-service-interface ams3.24
set services service-set Test Zone3 next-hop-service outside-service-interface ams3.25
set services web-filter multi-tenant-support
set services web-filter multi-tenant-hash file-hash-key ascii-text "$9$VjsgJikP36AGD6Ap0hcbs2"
set services web-filter multi-tenant-hash hash-method hmac-sha2-256
set services web-filter profile Test-Profile-3-Zone3 feed-name abc
set services web-filter profile Test-Profile-3-Zone3 global-dns-filter-stats-log-timer 20
set services web-filter profile Test-Profile-3-Zone3 dns-filter statistics-log-timer 5
set services web-filter profile Test-Profile-3-Zone3 dns-filter dns-resp-ttl 100
set services web-filter profile Test-Profile-3-Zone3 dns-filter wildcarding-level 10
  set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 inactive: client-interfaces xe-7/0/2.32
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 inactive: server-interfaces xe-7/2/0.36
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 inactive: client-routing-instance client_vr4
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 inactive: server-routing-instance server_vr4
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 term Test-Profile-3-Zone3-Area1-Customer1 feed-name customer2
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
```

```

Area1 term Test-Profile-3-Zone3-Area1-Customer1 from src-ip-prefix 10.12.1.1
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 term Test-Profile-3-Zone3-Area1-Customer1 then dns-sinkhole
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 term Test-Profile-3-Zone3-Area1-Customer2 feed-name customer2
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 term Test-Profile-3-Zone3-Area1-Customer2 from src-ip-prefix 2001:db8::0/96
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 term Test-Profile-3-Zone3-Area1-Customer2 then dns-sinkhole
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 term Test-Profile-3-Zone3-Area1-Customer3 from src-ip-prefix 2001:db8:bbbb::/96
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area1 term Test-Profile-3-Zone3-Area1-Customer3 then dns-sinkhole
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area2 inactive: client-interfaces xe-7/0/2.32
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area2 inactive: server-interfaces xe-7/2/0.36
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area2 inactive: client-routing-instance client_vr4
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area2 inactive: server-routing-instance server_vr4
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area2 term Test-Profile-3-Zone3-Area2-Customer1 from src-ip-prefix 22.21.128.0/17
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone3-
Area2 term Test-Profile-3-Zone3-Area2-Customer1 then dns-sinkhole
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone4-
Area2 feed-name customer2
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone4-
Area2 inactive: client-routing-instance client_vr4
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone4-
Area2 inactive: server-routing-instance server_vr4
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone4-
Area2 term Test-Profile-3-Zone4-Area2-Customer1 from src-ip-prefix 2001:0db8:0001:/48
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone4-
Area2 term Test-Profile-3-Zone4-Area2-Customer1 then dns-sinkhole
set services web-filter profile Test-Profile-3-Zone3 dns-filter-template Test-Profile-3-Zone4-
Area2 term wildcard then dns-sinkhole
set interfaces xe-7/0/0 unit 0 family inet address 10.11.1.1/24
set interfaces xe-7/0/1 unit 0 family inet address 10.12.1.1/24
set interfaces xe-7/0/2 flexible-vlan-tagging
set interfaces xe-7/0/2 mtu 9192
set interfaces xe-7/0/2 encapsulation flexible-ethernet-services
set interfaces xe-7/0/2 unit 1 vlan-id 10

```

```

set interfaces xe-7/0/2 unit 1 family inet address 198.31.100.1/24
set interfaces xe-7/0/2 unit 31 vlan-id 31
set interfaces xe-7/0/2 unit 31 family inet address 198.51.70.1/24;
set interfaces xe-7/0/2 unit 31 family inet6 address 2001:db8:10::0/96
set interfaces xe-7/0/2 unit 32 vlan-id 32
set interfaces xe-7/0/2 unit 32 family inet address 198.51.71.1/24;
set interfaces xe-7/0/2 unit 32 family inet6 address 2001:db8:11::0/96
set interfaces xe-7/0/2 unit 33 vlan-id 33
set interfaces xe-7/0/2 unit 33 family inet address 198.51.72.1/24
set interfaces xe-7/0/2 unit 33 family inet6 address 2001:db8:12::0/96
set interfaces xe-7/0/2 unit 34 vlan-id 34
set interfaces xe-7/0/2 unit 34 family inet address 198.51.73.1/24
set interfaces xe-7/0/2 unit 34 family inet6 address 2001:db8:13::0/96
set interfaces xe-7/0/2 unit 35 vlan-id 35
set interfaces xe-7/0/2 unit 35 vlan-id 35 family inet address 198.51.74.1/24
set interfaces xe-7/0/2 unit 3135 vlan-id 35 family inet6 address 2001:db8:14::0/96
set interfaces xe-7/0/2 unit 36 vlan-id 36
set interfaces xe-7/0/2 unit 36 family inet address 198.51.75.1/24
set interfaces xe-7/0/2 unit 36 family inet6 address 2001:db8:15::0/96
set interfaces xe-7/0/2 unit 37 vlan-id 37
set interfaces xe-7/0/2 unit 37 family inet address 198.51.76.1/24
set interfaces xe-7/0/2 unit 37 family inet6 address 2001:db8:16::0/96
set interfaces xe-7/0/2 unit 38 vlan-id 38
set interfaces xe-7/0/2 unit 38 family inet address 198.51.77.1/24
set interfaces xe-7/0/2 unit 38 family inet6 address 2001:db8:17::0/96
set interfaces xe-7/0/2 unit 39 vlan-id 39
set interfaces xe-7/0/2 unit 39 family inet address 198.51.78.1/24
set interfaces xe-7/0/2 unit 39 family inet6 address 2001:db8:18::0/96
set interfaces xe-7/0/2 unit 40 vlan-id 40
set interfaces xe-7/0/2 unit 40 family inet address 198.51.79.1/24
set interfaces xe-7/0/2 unit 40 family inet6 address 2001:db8:19::0/96
set interfaces xe-7/0/2 unit 41 vlan-id 41
set interfaces xe-7/0/2 unit 41 family inet address 198.51.80.1/24
set interfaces xe-7/0/2 unit 41 family inet6 address 2001:db8:20::0/96
set interfaces xe-7/2/0 flexible-vlan-tagging
set interfaces xe-7/2/0 mtu 1514
set interfaces xe-7/2/0 encapsulation flexible-ethernet-services
set interfaces xe-7/2/0 inactive unit 1 vlan-id 1
set interfaces xe-7/2/0 inactive unit 1 family inet address 198.168.50.0/24
set interfaces xe-7/2/0 inactive unit 1 family inet6 address 2001:0db0:1600:0::1/112
set interfaces xe-7/2/0 unit 2 vlan-id 2
set interfaces xe-7/2/0 unit 2 vlan-id 2 family inet address 198.100.70.0/24
set interfaces xe-7/2/0 unit 31 vlan-id 31

```

```

set interfaces xe-7/2/0 unit 31 family inet address 10.1.0.1/16
set interfaces xe-7/2/0 unit 31 family inet6 address 2001:0db0:1601:0::1/112
set interfaces xe-7/2/0 unit 32 vlan-id 32
set interfaces xe-7/2/0 unit 32 family inet address 10.2.0.1/16
set interfaces xe-7/2/0 unit 32 family inet6 address 2001:0db0:1602:0::1/112
set interfaces xe-7/2/0 unit 33 vlan-id 33
set interfaces xe-7/2/0 unit 33 family inet address 10.3.0.1/16
set interfaces xe-7/2/0 unit 33 vlan-id 33 family inet6 address 2001:0db0:1603:0::1/112
set interfaces xe-7/2/0 unit 34 vlan-id 34
set interfaces xe-7/2/0 unit 34 family inet address 10.0.0.1/16
set interfaces xe-7/2/0 unit 34 family inet6 address 2001:0db0:1600:0::1/112
set interfaces xe-7/2/0 unit 35 vlan-id 35
set interfaces xe-7/2/0 unit 35 family inet address 10.4.0.1/16
set interfaces xe-7/2/0 unit 35 family inet6 address 2001:0db0:1604:0::1/112
set interfaces xe-7/2/0 unit 36 vlan-id 36
set interfaces xe-7/2/0 unit 36 family inet address 10.5.0.1/16
set interfaces xe-7/2/0 unit 36 family inet6 address 2001:0db0:1605:0::1/112
set interfaces xe-7/2/0 unit 37 vlan-id 37
set interfaces xe-7/2/0 unit 37 family inet address 10.6.0.1/16
set interfaces xe-7/2/0 unit 37 family inet6 address 2001:0db0:1606:0::1/112
set interfaces xe-7/2/0 unit 38 vlan-id 38
set interfaces xe-7/2/0 unit 38 family inet address 10.7.0.1/16
set interfaces xe-7/2/0 unit 38 vlan-id 38 family inet6 address 2001:0db0:1607:0::1/112
set interfaces ams3 load-balancing-options member-interface mams-3/0/0
set interfaces ams3 load-balancing-options member-interface mams-3/1/0
set interfaces ams3 load-balancing-options member-failure-options redistribute-all-traffic
enable-rejoin
set interfaces ams3 load-balancing-options high-availability-options many-to-one preferred-
backup mams-3/1/0
set interfaces ams3 unit 22 family inet
set interfaces ams3 unit 22 family inet6
set interfaces ams3 unit 22 service-domain inside
set interfaces ams3 unit 22 load-balancing-options hash-keys ingress-key (source-ip destination-
ip )
set interfaces ams3 unit 24 family inet
set interfaces ams3 unit 24 family inet6
set interfaces ams3 unit 24 service-domain inside
set interfaces ams3 unit 24 family inet6 load-balancing-options hash-keys ingress-key (source-
ip destination-ip)
set interfaces ams3 unit 25 family inet
set interfaces ams3 unit 25 family inet6
set interfaces ams3 unit 25 service-domain inside
set interfaces ams3 unit 25 load-balancing-options hash-keys ingress-key (source-ip destination-

```

```

ip )
set routing-instances client_vr4 instance-type virtual-router
set routing-instances client_vr4 routing-options rib client_vr4.inet6.0 static route
2001:0db0:bbbb:0::0/49 next-hop 2001:0db0:7070:71::2
set routing-instances client_vr4 routing-options rib client_vr4.inet6.0 static route
2001:0db0:aaaa:8000::0/49 next-hop 2001:0db0:7070:71::3
set routing-instances client_vr4 routing-options rib client_vr4.inet6.0 static route 60::0/64
next-hop ams3.24
set routing-instances client_vr4 routing-options static route 10.12.1.1 next-hop 192.168.1.2
set routing-instances client_vr4 routing-options static route 22.21.128.0/17 next-hop 192.168.1.3
set routing-instances client_vr4 routing-options static route 0.0.0.0/0 next-hop ams3.24
set routing-instances client_vr4 routing-options static route 10.11.10.10/16 next-hop 192.168.1.4
set routing-instances client_vr4 routing-options static route 10.10.23.10/16 next-hop 192.168.1.5
set routing-instances client_vr4 routing-options static route 10.1.0.0/16 next-hop 192.168.1.6
set routing-instances client_vr4 routing-options static route 10.20.20.0/16 next-hop 192.168.1.7
set routing-instances client_vr4 routing-options static route 10.2.0.0/16 next-hop 192.168.1.8
set routing-instances client_vr4 routing-options static route 10.30.20.0/16 next-hop 192.168.1.9
set routing-instances client_vr4 routing-options static route 10.3.0.0/16 next-hop 192.168.10.
set routing-instances client_vr4 routing-options static route 10.40.20.0/16 next-hop 192.168.1.11
set routing-instances client_vr4 routing-options static route 10.4.0.0/16 next-hop 192.168.1.12
set routing-instances client_vr4 routing-options static route 10.50.20.0/16 next-hop 192.168.1.13
set routing-instances client_vr4 interface xe-7/0/0.0
set routing-instances client_vr4 interface xe-7/0/2.32
set routing-instances client_vr4 interface ams3.24
set routing-instances server_vr4 instance-type virtual-router
set routing-instances server_vr4 routing-options rib server_vr4.inet6.0 static route
2001:0db0:2221:0::0/48 next-hop ams3.25
set routing-instances server_vr4 routing-options rib server_vr4.inet6.0 static route
2001:db8:ffff::1/128 next-hop 2001:0db0:1605:0::2
set routing-instances server_vr4 routing-options rib server_vr4.inet6.0 static route
2001:db8:bbbb::1/128 next-hop 2001:0db0:1605:0::3
set routing-instances server_vr4 routing-options static route 10.10.20.1 next-hop ams3.25
set routing-instances server_vr4 routing-options static route 60.0.6.0/24 next-hop 192.0.2.2
set routing-instances server_vr4 routing-options static route 60.0.18.0/24 next-hop 192.0.2.3
set routing-instances server_vr4 routing-options static route 10.9.9.0/24 next-hop ams3.25
set routing-instances server_vr4 routing-options static route 60.0.19.0/24 next-hop 192.0.2.4
set routing-instances server_vr4 routing-options static route 60.0.20.0/24 next-hop 192.0.2.5
set routing-instances server_vr4 routing-options static route 60.0.21.0/24 next-hop 192.0.2.6
set routing-instances server_vr4 routing-options static route 60.0.22.0/24 next-hop 192.0.2.7
set routing-instances server_vr4 routing-options static route 60.0.23.0/24 next-hop 192.0.2.8
set routing-instances server_vr4 routing-options static route 60.0.24.0/24 next-hop 192.0.2.9
set routing-instances server_vr4 routing-options static route 60.0.25.0/24 next-hop 192.0.2.10
set routing-instances server_vr4 routing-options static route 60.0.26.0/24 next-hop 192.0.2.11

```

```
set routing-instances server_vr4 routing-options static route 60.0.27.0/24 next-hop 192.0.2.12
set routing-instances server_vr4 routing-options static route 60.0.28.0/24 next-hop 192.0.2.13
set routing-instances server_vr4 routing-options static route 10.1.0.0/16 next-hop ams3.25
set routing-instances server_vr4 interface xe-7/0/1.0
set routing-instances server_vr4 interface xe-7/2/0.36
set routing-instances server_vr4 interface ams3.25
set routing-options static route 0.0.0.0/0 next-hop 10.48.179.254
```

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2, you can configure DNS filtering if you are running Next Gen Services with the MX-SPC3 services card. Next Gen Services are supported on MX240, MX480 and MX960 routers.

DNS Request Filtering System Logging Error Messages

IN THIS SECTION

- [System Logging for DNS Request Filtering Overview | 404](#)
- [DNS Match-Event Syslog Format | 405](#)
- [Reason Mask Values & Interpretations for DNS Filtering | 407](#)
- [Per-Term Statistics Syslog Format | 409](#)
- [DNS Filtering Disallow-List File Add/Change Syslog Format | 411](#)
- [DNS Filtering Summary Report Statistics Syslog Format | 412](#)
- [DNS Filtering Per-Client-IP Statistics Syslog Format | 413](#)

The message format for system logs related to DNS request filtering differs slightly for the Next Gen Services MX-SPC3 services card versus early services cards. This topic describes the differences in the DNS request filtering related system log messages and provides a description of all fields in these messages.

System Logging for DNS Request Filtering Overview

Next Gen Services DNS request filtering system logging generates these events:

1. DNS match events (DNS_SR_MATCH_EVENT)
 - a. A single syslog is generated for each DNS match to the list of filtered domains.
2. Per-term statistics (DNS_SR_CUSTOMER_STATS)
 - a. Each term in the template represents a customer, enabling you to collect per-customer statistics.
 - b. You can configure the interval in which you want to collect statistics in each template.
3. You can report an event each time a DNS disallow-list file is added or updated (DNS_SR_FILE_UPDATE_NOTICE)
4. You can collect per-PIC Summary report statistics (DNS_SR_REPORT_STATS)
 - a. Statistics are generated every 5 minutes. This interval value is not configurable.
 - b. These stats are generated per-PIC basis.



NOTE: To enable these logs you must configure a syslog for each service-set for which you've configured dns-filtering.
All system log messages for Next Gen Services are configured at the service-set level using the following statement:

```
user@host# edit services service-set service-set-name syslog
```

To collect DNS request filtering system log messages, include `ur1f` in the `local-category` statement:

```
[edit services service-set ss1 syslog]
user@host# set local-category ur1f
```

5. You can collect per-client IP statistics (DNS_SR_CLIENT_IP_STATS)
 - a. This statistics are generated per-profile.
 - b. The interval for collecting these statistics is configurable per-profile.

DNS Match-Event Syslog Format



NOTE: System system log messages for Next Gen Services DNS request filtering doesn't include the FPC slot/PIC slot and UTC time.

Table 42 on page 405 describes the fields contained in DNS request filtering match events.

Table 42: DNS-Match-Event Syslog Format

Field Name	Description	Example
Time Stamp	Time when log entry was generated	Oct 27 10:04:19
Router Name	Host name of the router generating the record	Jnpr-router-01
Log Handle	Log handle to identify the log category	junos-url-filter
Match	Indicates a DNS match was detected.	JSERVICES_URLF_MATCH_EVENT: DNS_SR_MATCH_EVENT
Tag	Log-prefix configured	Tag=<value>
svc-set-name	Service-set name	svc-set-name=<value>
ID	ID assigned to the domain name (Size of ID is assumed to be a 32-bit number)	ID=12345
IP_Src	Source IP	IP_Src=10.1.5.72
IP_Dst	Destination IP (DNS resolver)	IP_Dst=10.1.1.10
Src_Prt	Source Port	Src_Prt=37344

Table 42: DNS-Match-Event Syslog Format (Continued)

Field Name	Description	Example
Dst_Prt	Destination Port	Dst_Prt=53
Sinkhole_IP	IP of sinkhole server from Domain Name Input List	Sinkhole_IP=10.1.50.64
Sinkhole_IPv6	IP of IPv6 sinkhole server from Domain Name Input List	Sinkhole_IPv6=2001:db8: 1003:1004:1005:1006:1007:1008
Sinkhole_fqdn	Sinkhole FQDN	Sinkhole_fqdn=NA
Count	Counter for match events to accommodate identical event records	Count=54
Replaced	Designates replacement of response domain (i.e. sinkholing)	Replaced=Y
Reason_Mask	Reason for action (if Replaced=N) [See table below for bit position enumeration]	Reason_Mask=0x0
QType	Query Type of the DNS request (A, AAAA, MX, CNAME, SRV, TXT)	QType=A
Profile	Profile Name [The Web filter profile name as configured]	Profile=profile_01

Table 42: DNS-Match-Event Syslog Format (Continued)

Field Name	Description	Example
Template	Template Name [The DNS filter template name as configured]	Template=template_01
Term	Term Name [The DNS filter term name as configured]	Term=term_01
Time	UNIX timestamp	Time=Wed Dec 20 12:25:24 2017

Here's an example of MX-SPC3 DNS filtering syslog format:

```
Feb 20 17:06:36 ce-bras-mx480-o junos-url-filter: JSERVICES_URLF_MATCH_EVENT: DNS_SR_MATCH_EVENT, Tag=tag, svc-set-name= s1, ID=1235, IP_SRC=10.2.2.3, IP_DST=10.101.10.100, SRC_PRT=34342, DST_PRT=53, Sinkhole_IP=10.1.1.1, Sinkhole_IPv6=NA, Sinkhole_fqdn=NA, Count=9, Replaced=Y, Reason_Mask=0x0, QType=A, Profile=webf-prof-1, Template=dnsf-temp-1, Term=dnsf-term-1, Time=Tue Jan 23 13:45:52 2018
```

Here's an example of MS-MPC DNS filtering syslog format:

```
Jan 23 13:45:52 cliq (FPC Slot 1, PIC Slot 1) 2018-01-23 21:45:52: {s1}[jservices-urlf]: JSERVICES_URLF_MATCH_EVENT: DNS_SR_MATCH_EVENT ID=1235, IP_SRC=10.2.2.3, IP_DST=10.101.10.100, SRC_PRT=34342, DST_PRT=53, Sinkhole_IP=10.1.1.1, Sinkhole_IPv6=NA, Sinkhole_fqdn=NA, Count=9, Replaced=Y, Reason_Mask=0x0, QType=A, Profile=webf-prof-1, Template=dnsf-temp-1, Term=dnsf-term-1, Time=Tue Jan 23 13:45:52 2018
```

Reason Mask Values & Interpretations for DNS Filtering

[Table 43 on page 407](#) describes the reason mask value fields and interpretations for MX Next Gen Services DNS filtering.

Table 43: Reason Mask Values & Interpretations for DNS Filtering

Bit Position	Hex Value	Interpretation	Additional Comments
	0x0	Replaced	

Table 43: Reason Mask Values & Interpretations for DNS Filtering (Continued)

Bit Position	Hex Value	Interpretation	Additional Comments
0	0x1	Reason Other	<i>Examples:</i> Fragmented packets, malformed packets
1	0x2	Not a supported DNS request type	<i>Examples:</i> SRV, TXT
2	0x4	Indicator action set to "Report-Only"	This is to enable testing of new indicators before putting them into Production.
3	0x8	Replace A/AAAA record error	
4	0x10	Replacement information not available	The domain name entry is marked "replace" but the sinkhole-ip/sinkhole-ipv6/sinkhole-fqdn is not provided.

Here's an example of MX Next Gen Services syslog format for DNS filtering showing the reason mask and interpretation:

```
Feb 20 17:06:36 ce-bras-mx480-o junos-url-filter: JSERVICES_URLF_MATCH_EVENT: DNS_SR_MATCH_EVENT, Tag=tag, svc-set-name= s1, ID=1235, IP_SRC=10.2.2.3, IP_DST=10.101.10.100, SRC_PRT=34342, DST_PRT=53, Sinkhole_IP=10.1.1.1, Sinkhole_IPv6=NA, Sinkhole_fqdn=NA, Count=9, Replaced=Y, Reason_Mask=0x0, QType=A, Profile=webf-prof-1, Template=dnsf-temp-1, Term=dnsf-term-1, Time=Tue Jan 23 13:45:52 2018
```

Here's an example of MS-MPC DNS filtering syslog format:

```
Jan 23 13:45:52 cliq (FPC Slot 1, PIC Slot 1) 2018-01-23 21:45:52: {s1}[jservices-urlf]: JSERVICES_URLF_MATCH_EVENT: DNS_SR_MATCH_EVENT ID=1235, IP_SRC=10.2.2.3, IP_DST=10.101.10.100, SRC_PRT=34342, DST_PRT=53, Sinkhole_IP=10.1.1.1, Sinkhole_IPv6=NA, Sinkhole_fqdn=NA, Count=9, Replaced=Y, Reason_Mask=0x0, QType=A, Profile=webf-prof-1, Template=dnsf-temp-1, Term=dnsf-term-1, Time=Tue Jan 23 13:45:52 2018
```

Per-Term Statistics Syslog Format

Table 44 on page 409 describes the fields for MX Next Gen Services DNS filtering per-term statistics syslog format.

Table 44: Per-Term Statistics Syslog Format

Field Name	Description	Example
Time Stamp	Time when log entry was generated	Oct 27 10:04:17
Router Name	Host name of the router generating the record	Jnpr-router-01
Log Handle	Log handle to identify the log category	junos-url-filter
Match	A term(customer) statistics record	JSERVICES_URLF_CUSTOMER_STAT S: DNS_SR_CUSTOMER_STATS
Tag	Log-prefix configured	Tag=<value>
svc-set-name	Service-set name	svc-set-name=<value>
Profile	Profile Name [The Web filter profile name as configured]	Profile=profile_01
Template	Template Name [The DNS filter template name as configured]	Template=template_01
Term	Term Name [The DNS filter term name as configured]	Term=term_01

Table 44: Per-Term Statistics Syslog Format (Continued)

Field Name	Description	Example
Packets_Processed	Total DNS Requests Processed	Requests_Processed=200
DNS_UDP_Packets_Processed	DNS UDP Requests Processed	DNS_UDP_Requests_Processed=98
DNS_TCP_Packets_Processed	DNS TCP Requests Processed	DNS_TCP_Requests_Processed=35
DNS_UDP_Requests_sinkholed	DNS UDP Requests sink-holed	DNS_UDP_Requests_Sinkholed =50
DNS_TCP_Requests_sinkholed	DNS TCP Requests sink-holed	DNS_TCP_Requests_Sinkholed =50
DNS_UDP_Requests_reported	DNS UDP Requests reported	DNS_UDP_Requests_Reported =50
DNS_TCP_Requests_reported	DNS TCP Requests reported	DNS_TCP_Requests_Reported =50
Time	UNIX timestamp	Time=Wed Dec 20 12:25:24 2017
Count	Counter to accommodate identical event records	Count=10

Here's an example of MX-SPC3 DNS filtering syslog format for per-term statistics:

```
Feb 25 14:25:45 curve junos-url-filter: JSERVICES_URLF_CUSTOMER_STATS: DNS_SR_CUSTOMER_STATS, Tag , svc-set-name
s1, Profile=DNS_CUSTOMER-A, Template=DNS_CUSTOMER-A, Term=DNS_CUSTOMER-A, Requests_Processed=0,
DNS_UDP_Requests_Processed=0, DNS_TCP_Requests_Processed=0, DNS_UDP_Requests_Sinkholed=0,
DNS_TCP_Requests_Sinkholed=0, DNS_UDP_Requests_Reported=0, DNS_TCP_Requests_Reported=0, Time=Mon Feb 25 14:25:45
2019, Count=13
```

Here's an example of MS-MPC DNS filtering syslog format:

```
Mar 8 12:16:05 iphone3gs (FPC Slot 5, PIC Slot 0) 2019-03-08 20:16:04: {ATT-Zone5}[jservices-urlf]:
JSERVICES_URLF_CUSTOMER_STATS: DNS_SR_CUSTOMER_STATS, Profile=ATT-Profile-5-Zone5, Template=ATT-Profile-5-Zone5-
Area1, Term=ATT-Profile-5-Zone5-Area1-Customer3, Requests_Processed=0, DNS_UDP_Requests_Processed=0,
DNS_TCP_Requests_Processed=0, DNS_UDP_Requests_Sinkholed=0, DNS_TCP_Requests_Sinkholed=0,
DNS_UDP_Requests_Reported=0, DNS_TCP_Requests_Reported=0, Time=Fri Mar 08 12:16:05 2019, Count=111
```

DNS Filtering Disallow-List File Add/Change Syslog Format

Table 45 on page 411 describes the fields for MX Next Gen Services DNS filtering disallow-list file additions and updates syslog format.

Table 45: Disallow-List File Add/Change Syslog Format

Field Name	Description	Example
Time Stamp	Time when log entry was generated	Oct 27 10:04:17
Router Name	Host name of the router generating the record	Jnpr-router-01
Log Handle	Log handle to identify the log category	junos-url-filter
Match	The domain disallow-list file updated for the template. .	JSERVICES_URLF_FILE_UPDATE_NOTICE : DNS_SR_FILE_UPDATE_NOTICE
Tag	Log-prefix configured	Tag=<value>
svc-set-name	Service-set name	svc-set-name=<value>
File Name	Name of the file	File_Name=shdb.txt
File Version	Version of the file	File_Version=20170314_01
Updated	File Update Time	Domain_Filter_File_Updated=Fri Oct 27 10:56:42 2017
Profile	Profile Name [The Web filter profile name as configured]	Profile=profile_01

Table 45: Disallow-List File Add/Change Syslog Format (Continued)

Field Name	Description	Example
Template	Template Name [The DNS filter template name as configured]	Template=template_01
Domains	Number of Domains in the file	Domains=12
Report-Only-Domains	Number of Report-Only domains in the file	Report_Only_Domains=3

Here's an example of the syslog format for MX-SPC3 DNS filtering disallow-list add/change file updates:

```
Feb 25 14:36:47 curve junos-url-filter: JSERVICES_URLF_FILE_UPDATE_NOTICE: DNS_SR_FILE_UPDATE_NOTICE, Tag=, svc-set-name=s1, File_Name=test_dns_sink.txt, File_Version=20180911_01, Domain_Filter_File_Updated=Mon Feb 25 14:36:47 2019 Profile=DNS_CUSTOMER-A, Template=DNS_CUSTOMER-A, Domains=18, Report_Only_Domains=0
```

Here's an example of the syslog format for DNS filtering disallow-list file changes with the MS-MPC services card:

```
Jan 23 13:34:34 cliq (FPC Slot 1, PIC Slot 1) 2018-01-23 21:34:33: {s1}[jservices-urlf]: JSERVICES_URLF_FILE_UPDATE_NOTICE: DNS_SR_FILE_UPDATE_NOTICE, File_Name=dnsf1_hashed.txt, File_Version=20170314_01, Domain_Filter_File_Updated=Tue Jan 23 13:34:34 2018 Profile=webf-prof-1, Template=dnsf-temp-1, Domains=4, Report_Only_Domains=1
```

DNS Filtering Summary Report Statistics Syslog Format

Summary report statistics syslog format Stats will be reported in syslog with the following format:

Here's an example summary report syslog message for MX-SPC3 Next Gen Services DNS filtering:

```
Feb 25 11:50:39 curve junos-url-filter: JSERVICES_URLF_REPORT_STATS: DNS_SR_REPORT_STATS, Tag=, svc-set-name=s1, TCP_DNS_Packets=0, TCP_DNS_Non_Segmented=0, TCP_DNS_Segmented=0, Count=1
```

Here's an example summary report syslog message for MS-MPC services card DNS filtering:

```
Mar 8 12:20:41 iphone3gs (FPC Slot 5, PIC Slot 1) 2019-03-08 20:20:40: {ATT-Zone1}[jservices-urlf]: JSERVICES_URLF_REPORT_STATS: DNS_SR_REPORT_STATS, TCP_DNS_Packets=0, TCP_DNS_Non_Segmented=0, TCP_DNS_Segmented=0, Count=169
```

DNS Filtering Per-Client-IP Statistics Syslog Format

Table 46 on page 413 describes the syslog fields for MX-SPC3 DNS filtering per-client-IP statistics that is reported per-PIC, per-profile for all known client IP addresses known to the system.

Table 46: Per-Client-IP Statistics Syslog Format

Field Name	Description	Example
Time Stamp	Time when log entry was generated	Oct 27 10:04:17
Router Name	Host name of the router generating the record	Jnpr-router-01
Log Handle	Log handle to identify the log category	junos-url-filter
Match	Log for per-Client IP stats	JSERVICES_URLF_CLIENT_IP_STATS: DNS_SR_CLIENT_IP_STATS
Tag	Log-prefix configured	Tag=<value>
svc-set-name	Service-set name	svc-set-name=<value>
Client-IP	IP address of the client	Client-IP=10.1.1.1
Profile	Profile Name [The Web filter profile name as configured]	Profile=profile_01
Template	Template Name [The DNS filter template name as configured]	Template=template_01

Table 46: Per-Client-IP Statistics Syslog Format (Continued)

Field Name	Description	Example
Term	Term Name [The DNS filter term name as configured]	Term=term_01
A_Req	DNS A-Record Requests Processed	A_Req=10
AAAA_Req	DNS AAAA-Record Requests Processed	AAAA_Req=10
MX_Req	DNS MX-Record Requests Processed	MX_Req=4
CNAME_Req	DNS CNAME-Record Requests Processed	CNAME_Req=4
SRV_Req	DNS SRV-Record Requests Processed	SRV_Req=4
TXT_Req	DNS TXT-Record Requests Processed	TXT_Req=4
ANY_Req	DNS ANY-Record Requests Processed	ANY_Req=4
A_Req_SH	DNS A-Record Requests sink-holed	A_Req_SH =5
AAAA_Req_SH	DNS AAAA-Record Requests sink-holed	AAAA_Req_SH=5
MX_Req_SH	DNS MX-Record Requests Sink-holed	MX_Req_SH=4

Table 46: Per-Client-IP Statistics Syslog Format (Continued)

Field Name	Description	Example
CNAME_Req_SH	DNS CNAME-Record Requests Sink-holed	CNAME_Req_SH=4
SRV_Req_SH	DNS SRV-Record Requests Sink-holed	SRV_Req_SH=4
TXT_Req_SH	DNS TXT-Record Requests Sink-holed	TXT_Req_SH=4
ANY_Req_SH	DNS ANY-Record Requests Sink-holed	ANY_Req_SH=4
Req_Rep	DNS Requests reported	Req_Rep=5

Here's an example per-client-IP-statistics for MX-SPC3 DNS filtering:

```
Feb 25 11:50:39 curve junos-url-filter: JSERVICES_URLF_CLIENT_IP_STATS: DNS_SR_CLIENT_IP_STATS, Tag=tag, svc-set-name=s1, Client-IP=10.2.2.3, Profile=webf-prof-1, Template=dnsf-temp-1, Term=dnsf-term-1, A_Req=0, AAAA_Req=0, MX_Req=0, CNAME_Req=0, SRV_Req=0, TXT_Req=0, ANY_Req=2, A_Req_SH=0, AAAA_Req_SH=0, MX_Req_SH=0, CNAME_Req_SH=0, SRV_Req_SH=0, TXT_Req_SH=0, ANY_Req_SH=0, Req_Rep=2
```

Here's an example syslog message for DNS filtering client-IP statistics on MS-MPC services cards:

```
Mar 7 17:58:54 iphone3gs (FPC Slot 5, PIC Slot 3) 2019-03-08 01:58:54: {dns}[jservices-urlf]: JSERVICES_URLF_CLIENT_IP_STATS: DNS_SR_CLIENT_IP_STATS, Client-IP=2008:db8:2228:8001::1, Profile=dns-profile1, Template=dns1, Term=3, A_Req=19, AAAA_Req=19, MX_Req=0, CNAME_Req=0, SRV_Req=0, TXT_Req=0, ANY_Req=0, A_Req_SH=19, AAAA_Req_SH=19, MX_Req_SH=0, CNAME_Req_SH=0, SRV_Req_SH=0, TXT_Req_SH=0, ANY_Req_SH=0, Req_Rep=0
```

7

PART

URL Filtering

- [URL Filtering | 417](#)
-

URL Filtering

IN THIS CHAPTER

- [URL Filtering Overview | 417](#)
- [Configuring URL Filtering | 423](#)

URL Filtering Overview

IN THIS SECTION

- [URL Filter Database File | 420](#)
- [URL Filter Profile Caveats | 421](#)

You can use URL filtering to determine which Web content is not accessible to users.

Components of this feature include the following:

- URL filter database file
- Configuration of one or more templates (up to eight per profile)
- URL Filter Plug-in (jservices-urlf)
- URL filtering daemon (url-filterd)

The URL filter database file is stored on the Routing Engine and contains all the disallowed URLs. Configured *templates* define which traffic to monitor, what criteria to match, and which actions to take. You configure the templates and the location of the URL filter database file in a *profile*.

Starting in Junos OS Release 17.2R2 and 17.4R1, for Adaptive Services, you can disable the filtering of HTTP traffic that contains an embedded IP address (for example, `http://10.1.1.1`) belonging to a

disallowed domain name in the URL filter database. Starting in Junos OS Release 19.3R2, this same functionality is supported for Next Gen Services on MX240, MX480, and MX960.

To enable the URL filtering feature, you must configure `jservices-urlf` as the *package-name* at the `[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]` hierarchy level. Once enabled, `jservices-urlf` maintains the URL filtering profile and receives all traffic to be filtered, the filtering criteria, and the action to be taken on the filtered traffic.



NOTE: MX-SPC3 does not explicitly need `jservices-urlf` as the *package-name* at the `[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]` hierarchy level. It is supported by default.

The URL filtering daemon (`url-filterd`), which also resides on the Routing Engine, resolves the domain name of each URL in the URL filter database to a list of IPv4 and IPv6 addresses. It then downloads the list of IP addresses to the service PIC, which runs `jservices-urlf`. Then `url-filterd` interacts with the Dynamic Firewall process (`dfwd`) to install filters on the Packet Forwarding Engine to punt the selected traffic from the Packet Forwarding Engine to the service PIC.

As new HTTP and HTTPS traffic reaches the router, a decision is made based on the information in the URL filter database file. The filtering rules are checked and either the router accepts the traffic and passes it on or blocks the traffic. If the traffic is blocked, one of the following configured actions is taken:

- An HTTP redirect is sent to the user.
- A custom page is sent to the user.
- An HTTP status code is sent to the user.
- A TCP reset is sent.

Accept is also an option. In this case, the traffic is not blocked.

[Figure 11 on page 419](#) illustrates the URL filtering for HTTP sessions.

Figure 11: Packet Flow-URL Filtering for HTTP Sessions

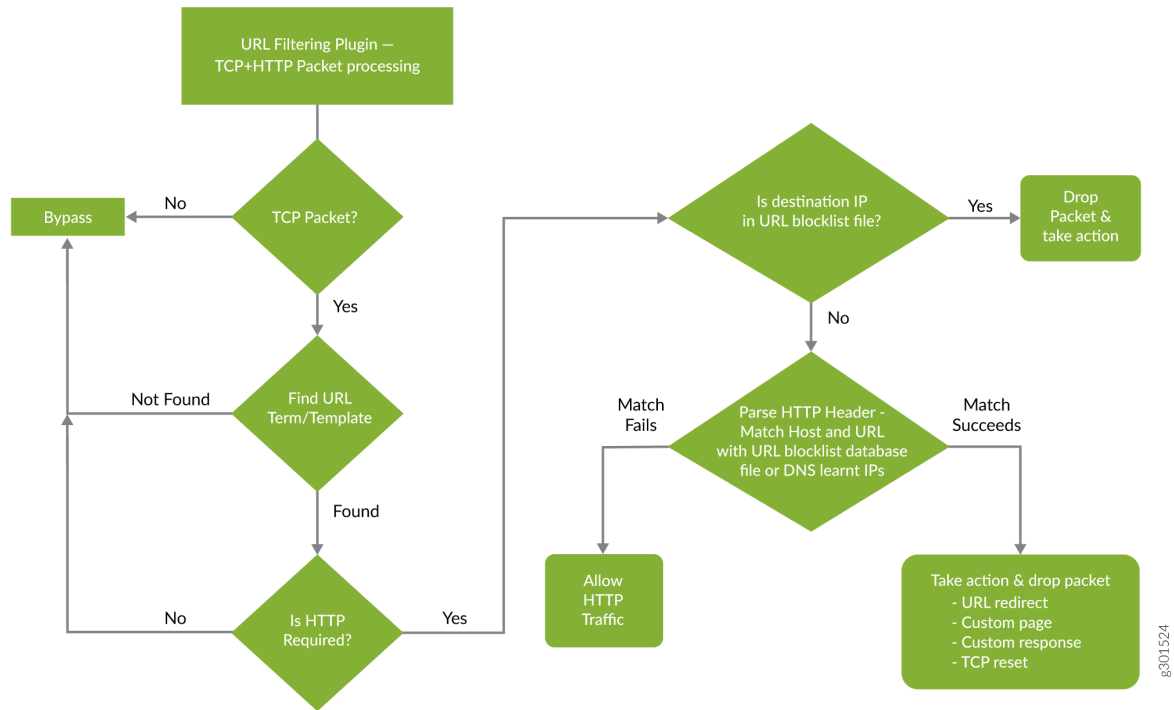
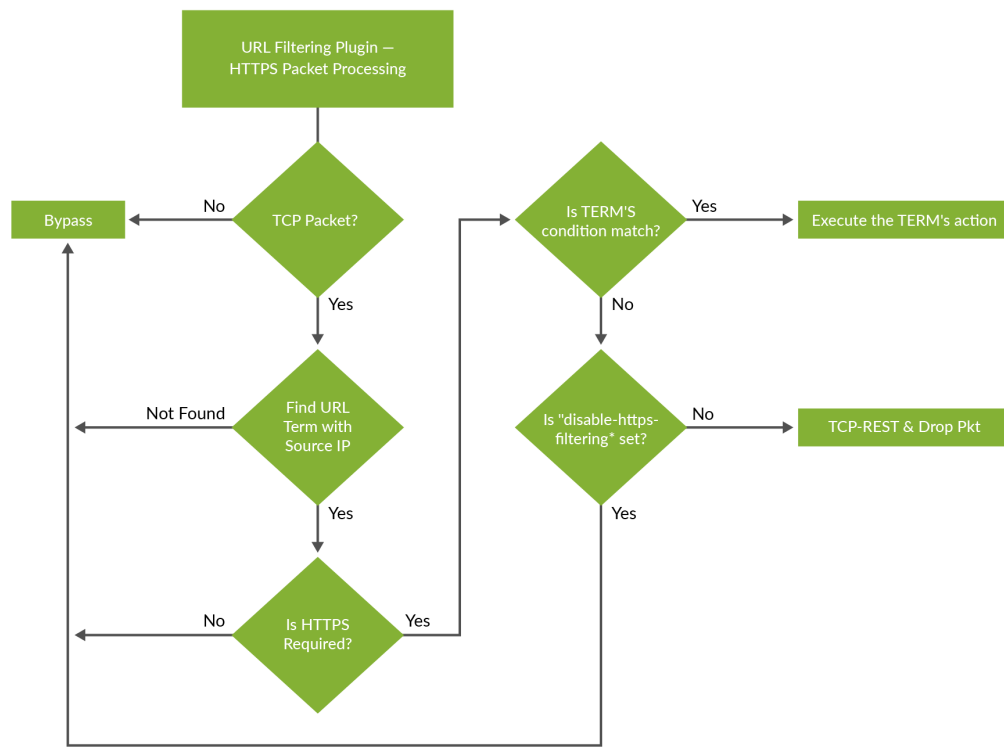


Figure 12 on page 420 illustrates the URL filtering for HTTPS sessions.

Figure 12: Packet Flow-URL Filtering for HTTPS Sessions



For more details on the URL filtering feature, see the following sections:

URL Filter Database File

The URL filter database file contains entries of URLs and IP addresses. Create the URL filter database file in the format indicated in [Table 47 on page 420](#) and locate it on the Routing Engine in the `/var/db/url-filterd` directory.

Table 47: URL Filter Database File Format

Entry	Description	Example
FQDN	Fully qualified domain name.	www.badword.com/jjj/bad.jpg

Table 47: URL Filter Database File Format (*Continued*)

Entry	Description	Example
URL	Full string URL without the Layer 7 protocol.	www.srch.com/*badword*/ www.srch.com www.srch.com/xyz www.srch.com/xyz*
IPv4 address	HTTP request on a specific IPv4 address.	10.1.1.199
IPv6 address	HTTP request on a specific IPv6 address.	1::1

You must specify a custom URL filter database in the profile. If needed, you can also assign a custom URL filter database file with any template, and that database takes precedence over the database configured at the profile level.

If you change the contents of the URL filter database file, use the request services (url-filter | web-filter) update command. Other commands to help maintain the URL filter database file include the following:

- request services (url-filter | web-filter) delete
- request services (url-filter | web-filter) force
- request services (url-filter | web-filter) validate

URL Filter Profile Caveats

The URL filter profile consists of from one to eight templates. Each template consists of a set of configured logical interfaces where traffic is monitored for URL filtering and one or more terms.

A *term* is a set of match criteria with actions to be taken if the match criteria is met. You must configure at least one term to configure URL filtering. Each term consists of a *from* statement and a *then* statement, where the *from* statement defines the source IP prefixes and destination ports that are monitored. The *then* statement specifies the action to be taken. If you omit the *from* statement, any source IP prefix and any destination port are considered to match. But you can omit only one *from* statement per template or per profile.

Example configuration of multiple terms without from statements

```
template1 {
  client-interfaces [ xe-4/0/3.35 xe-4/0/3.36 ];
  server-interfaces xe-4/0/0.31;
  dns-source-interface xe-4/0/0.1;
  dns-routing-instance data_vr;
  routing-instance data_vr2;
  dns-server 50.0.0.3;
  dns-retries 3;
  url-filter-database url_database.txt;
  term term1 {
    then {
      tcp-reset;
    }
  }
  term term2 {
    then {
      redirect-url www.google.com;
    }
  }
}
```

If you omit more than one from statement per template, you will get the following error message on commit:

```
URLFD_CONFIG_FAILURE: Configuration not valid:
Cannot have two wild card terms in template template1
error: configuration check-out failed
```

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2, this same functionaly is supported for Next Gen Services on MX240, MX480, and MX960.

17.2R2 Starting in Junos OS Release 17.2R2 and 17.4R1, for Adaptive Services, you can disable the filtering of HTTP traffic that contains an embedded IP address (for example, `http://10.1.1.1`) belonging to a disallowed domain name in the URL filter database.

RELATED DOCUMENTATION

request services url-filter update url-filter-database file

request services url-filter force dns-resolution

request services url-filter delete gencfg-data

request services url-filter validate

Configuring URL Filtering

To configure the URL filtering feature, you must first configure `jservices-urlf` as the *package-name* at the `[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]` hierarchy level. For more information on configuring the extension-provider `package package-name` configuration statement, see the *package (Loading on PIC)* statement.



NOTE: MX-SPC3 does not explicitly need `jservices-urlf` as the *package-name* at the `[edit chassis fpc slot-number pic pic-number adaptive-services service-package extension-provider]` hierarchy level. It is supported by default.

URL filtering is configured on a service PIC. The interfaces you are dealing with are services interfaces (which use the `ms` prefix) or aggregated multiservices (AMS) interfaces (which use the `ams` prefix). For more information on AMS interfaces, see the *Adaptive Services Interfaces User Guide for Routing Devices* starting with *Understanding Aggregated Multiservices Interfaces*.

A URL filtering *profile* is a collection of templates. Each template consists of a set of criteria that defines which URLs are disallowed and how the recipient is notified.

To configure the URL profile:

1. Assign a name to the URL profile.

```
[edit]
user@host# edit services (web-filter | url-filter) profile profile-name
```

Starting in Junos OS Release 18.3R1, for Adaptive Services, configure the profile at the [edit services web-filter] hierarchy level. Before Junos OS Release 18.3R1, configure the profile at the [edit services url-filter] hierarchy level. Starting in Junos OS Release 19.3R2, this same functionality is available for Next Gen Services on MX240, MX480, and MX960.

2. Specify the name of the URL filter database to use.

```
[edit services (web-filter | url-filter) profile profile-name]
user@host# set url-filter-database filename
```

3. Configure one or more templates for the profile.

To configure each template:

- a. Name the template.

```
[edit services (web-filter | url-filter) profile profile-name]
user@host# set (url-filter-template template-name | template template-name)
```



NOTE: Starting in Junos OS Release 18.3R1, configure the template with the url-filter-template statement. Before Junos OS Release 18.3R1, configure the template with the template statement.

- b. Go to that new template hierarchy level.

```
[edit services (web-filter | url-filter) profile profile-name]
user@host# edit (url-filter-template template-name | template template-name)
```

- c. Specify the name of the URL filter database to use.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set url-filter-database filename
```

- d. Specify the loopback interface for which the source IP address is picked for sending DNS queries.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set dns-source-interface loopback-interface-name
```

- e. Disable the filtering of HTTP traffic that contains an embedded IP address (for example, http://10.1.1.1) belonging to a disallowed domain name in the URL filter database.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set disable-url-filtering
```

- f. Configure the DNS resolution time interval in minutes.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set dns-resolution-interval minutes
```

- g. Configure the number of retries for a DNS query in case the query fails or times out.

```
[edit services (web-filter | url-filter) profile profile-name]
user@host# set dns-retries number
```

- h. Specify the IP addresses (IPv4 or IPv6) of DNS servers to which the DNS queries are sent.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set dns-server [ip-address]
```

- i. Specify the client-facing logical interfaces on which the URL filtering is configured.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set client-interfaces [ client-interface-name ]
```

- j. Specify the server-facing logical interfaces on which the URL filtering is configured.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set server-interfaces [ server-interface-name ]
```

- k. Specify the routing instance on which the URL filtering is configured.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set routing-instance routing-instance-name
```

- l. Specify the routing instance on which the DNS server is reachable.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# dns-routing-instance dns-routing-instance-name
```

4. Configure the term information.

Terms are used in filters to segment the policy or filter into small match and action pairs.

- a. Name the term.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# set term term-name
```

- b. Go to the new term hierarchy level.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name)]
user@host# edit term term-name
```

- c. Specify the source IP address prefixes for traffic you want to filter.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name) term term-name]
user@host# set from src-ip-prefix [prefix]
```

- d. Specify the destination ports for traffic you want to filter.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name) term term-name]
user@host# set from dest-port [port]
```

- e. Configure an action to take.

```
[edit services (web-filter | url-filter) profile profile-name (url-filter-template
template-name | template template-name) term term-name]
user@host# set then action
```

The action can be one of the following:

<code>custom-page <i>custom-page</i></code>	Send a custom page string to the user.
<code>http-status-code <i>http-status-code</i></code>	Send an HTTP status code to the user.
<code>redirect-url <i>redirect-url</i></code>	Send an HTTP redirect to the user.
<code>tcp-reset</code>	Send a TCP reset to the user.

5. Associate the URL profile with a next-hop service set.





NOTE: For URL filtering, you must configure the service set as a next-hop service set.

```
[edit]
user@host# set services service-set service-set-name (web-filter-profile profile-name | url-
filter-profile profile-name)
user@host# set services service-set service-set-name next-hop-service inside-service-
interface interface-name.unit-number
```



```
user@host# set services service-set service-set-name next-hop-service outside-service-  
interface interface-name.unit-number
```

 **NOTE:** The service interface can also be of the `ams` prefix. If you are using `ams` interfaces at the `[edit services service-set service-set-name]` hierarchy level for the URL filter, you must also configure the `load-balancing-options hash-keys` statement at the `[edit interfaces ams-interface-name unit number]` hierarchy level. .

 **NOTE:** Starting in Junos OS Release 18.3R1, configure the service set with the `web-filter-profile` statement. Before Junos OS Release 18.3R1, configure the service set with the `url-filter-profile` statement.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2, this same functionality is available for Next Gen Services on MX240, MX480, and MX960.
18.3R1	Starting in Junos OS Release 18.3R1, for Adaptive Services. configure the profile at the <code>[edit services web-filter]</code> hierarchy level. Before Junos OS Release 18.3R1, configure the profile at the <code>[edit services url-filter]</code> hierarchy level.

RELATED DOCUMENTATION

| *Configuring Service Sets to be Applied to Services Interfaces*

8

PART

Integration of Juniper ATP Cloud and Web filtering on MX Routers

- [Integration of Juniper ATP Cloud and Web filtering on MX Routers | 430](#)
-

Integration of Juniper ATP Cloud and Web filtering on MX Routers

IN THIS CHAPTER

- [Integration of Juniper ATP Cloud and Web Filtering on MX Series Routers | 430](#)

Integration of Juniper ATP Cloud and Web Filtering on MX Series Routers

IN THIS SECTION

- [Overview | 430](#)
- [Configuring the Web Filter Profile for Sampling | 436](#)
- [GeoIP Filtering | 441](#)
- [Global Allowlist and Global Blocklist | 443](#)

Overview

IN THIS SECTION

- [Benefits | 431](#)
- [Understanding Policy Enforcer and Juniper ATP Cloud | 431](#)
- [Security Intelligence \(SecIntel\) - Overview | 433](#)
- [Web Filtering \(URL-Filterd\) - Overview | 434](#)

Juniper Advanced Threat Prevention (Juniper ATP Cloud) is integrated with MX series routers to protect all hosts in your network against evolving security threats by employing cloud-based threat detection software with a next-generation firewall system.

This topic provides an overview of Juniper ATP Cloud, Policy Enforcer, Security Intelligence, Web filtering, and their benefits when integrated on MX Series routers .

For details on the platform and release information, see [Feature Explorer](#) .

Benefits

- Simplifies deployment and enhances the anti-threat capabilities when integrated with the MX routers.
- Delivers protection against “zero-day” threats using a combination of tools to provide robust coverage against sophisticated, evasive threats.
- Checks inbound and outbound traffic with policy enhancements that allow users to stop malware, quarantine infected systems, prevent data exfiltration, and disrupt lateral movement.
- Supports High Availability to provide uninterrupted service.
- Provides scalability to handle increasing loads that require more computing resources, increased network bandwidth to receive more customer submissions, and a large storage for malware.
- Provides deep inspection, actionable reporting, and inline malware blocking.
- Provides ability to provide tenancy information using VRF information in logs

Understanding Policy Enforcer and Juniper ATP Cloud

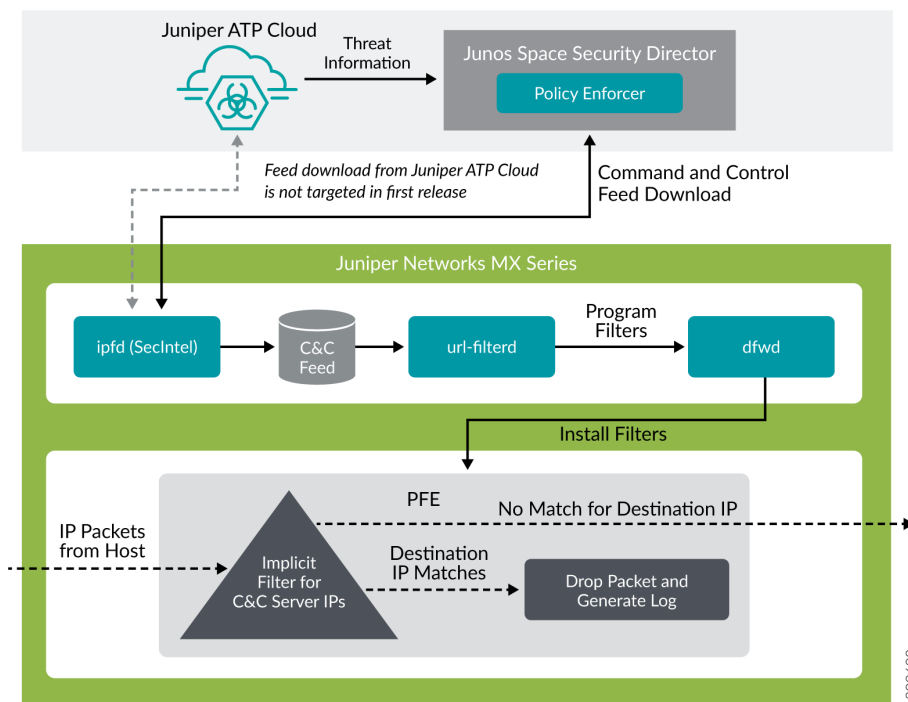
Juniper Networks Security Director comprises a feature called the Policy Enforcer (PE) that enables it to learn from threat conditions, automate the policy creation, and to dynamically deploy enforcement to Juniper devices in the network.

[Figure 13 on page 432](#) illustrates the traffic flow between the PE, the Juniper ATP Cloud, and the MX router which functions as a firewall.

- Policy Enforcer (PE) learns from threat conditions, automates the policy creation, and deploys enforcement to Juniper devices in the network.
- Juniper Advanced Threat Prevention (Juniper ATP Cloud) protects all hosts in your network by employing cloud-based threat detection software with a next-generation firewall system.
- MX router fetches the threat intelligence feeds from Policy Enforcer (PE) and implements those policies to quarantine compromised hosts. It comprises of the following important components:

- Security Intelligence process
- Web Filtering process
- Firewall process

Figure 13: System Architecture



To understand the functionality of the system architecture consider the following example—if a user downloads a file from the Internet and that file passes through an MX firewall, the file can be sent to the Juniper ATP Cloud cloud for malware inspection (depending on your configuration settings.) If the file is determined to be malware, PE identifies the IP address and MAC address of the host that downloaded the file. Based on a user-defined policy, that host can be put into a quarantine VLAN or blocked from accessing the Internet.

MX Series routers can be integrated with the Juniper ATP Cloud to prevent compromised hosts (botnets) from communicating with command and control servers:

- Starting in Junos OS Release 18.4R1 with the Adaptive Services as an Inline security capability
- Starting in Junos OS Release 19.3R2 with the Next Gen Services as an Inline security capability

MX Sseries routers can download C&C and Geo-IP from either of the following methods:

- Indirect method-Policy Enforcer acts like a feed proxy to all devices with in a defined environment. This method is beneficial in avoiding individual devices accessing Juniper ATP cloud services on the Internet. Thus, decreasing the vulnerability of the devices reaching out to Internet.
- Direct method-MX series routers enroll directly to Juniper ATP cloud to download the C&C and Geo-IP feeds.

Security Intelligence (SecIntel) - Overview

The Security Intelligence process (IPFD), is responsible for downloading the security intelligence feeds and parsing from the feed connector or ATP Cloud cloud feed server. The IPFD process on the MX platforms fetches the command and control IPv4/IPv6 feeds from Policy Enforcer. C&C feeds are essentially a list of servers that are known command and control servers for botnets. The list also includes servers that are known sources for malware downloads. The information thus fetched is saved in a file (`urlf_si_cc_db.txt`) created under the `/var/db/url-filterd` directory.

The file format of the disallowed IPs sent by IPFD to the web filtering process is as follows:

IPv4 address | IPv6 address, threat-level.

The *threat-level* is an integer ranging from 1 to 10 to indicate the threat level of files scanned for malware and for infected hosts. Here, 1 represents the lowest threat level and 10 represents the highest threat level.

For example: 178.10.19.20, 4

Here, 178.10.19.20 indicates the disallowed IP and 4 indicates the *threat-level*.

The C&C feed database is synced onto the backup Routing Engine. IPFD then shares the information to the web filtering process (`url-filterd`). The web filtering process reads the file contents and configures the filters accordingly.

Configuring Security Intelligence to Download the CC Feed from Policy Enforcer

To download the command and control IPv4/IPv6 feeds from Juniper ATP Cloud/Policy Enforcer, include the `security-intelligence` statement at the `[edit services]` hierarchy as shown in the following example:

```
security-intelligence {
  authentication {
    auth-token 7QGSBL5ZRKR5UHUZ2X2R6QLHB656D5EN;
  }
  url https://10.92.83.245:443/api/v1/manifest.xml;
  traceoptions {
```

```

file security-intelligence.log size 1g;
level all;
flag all;
}
}

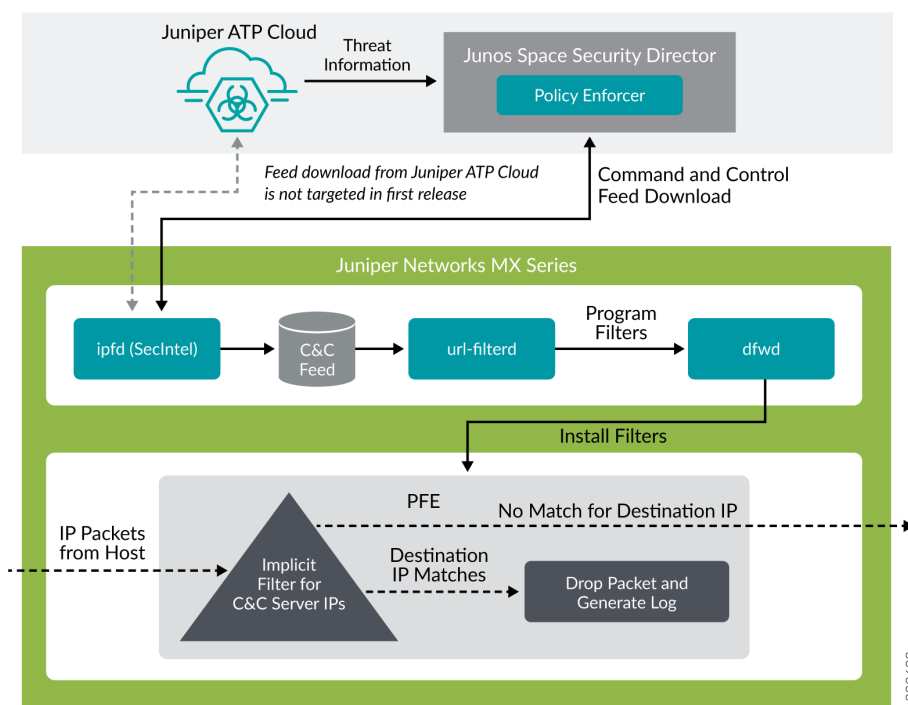
```

Web Filtering (URL-Filterd) - Overview

The web filtering process reads the file contents fetched from the IPFD and configures the filters on the Packet Forwarding Engine accordingly. The web filtering process enforces the command and control feeds by programming the filters in the Packet Forwarding Engine to block the packets destined to the blocked IP addresses and to generate logs for reporting the incident.

Figure 14 on page 434 illustrates the way C&C feed is fetched by the IPFD and then processed by the web filtering process.

Figure 14: Web Filtering



The web filter profile can have more than one templates. Each template consists of a set of configured logical interfaces for Web filtering and one or more terms. A term is a set of match criteria with actions to be taken if the match criteria is met. To configure the web filter profile to use dynamically fetched C&C feed, you can configure the security-intelligence-policy command under the [edit services web-filter

profile *profile-name* hierarchy level. You need not configure a term for a security-intelligence-policy based web filter profiles.

You can configure the following threat level actions for the web filter profile at the edit web-filter profile *profile-name* security-intelligence-policy threat-level *threat-level* threat-action hierarchy level:

- drop
- drop-and-log
- log

You can configure only one threat-action for each threat level. If the threat-action is not configured for a particular threat level, the default threat-action is accept.

Starting in Junos OS Release 24.4R1, for threats configured with log action, the threat-level and the tenant or the VRF information is embedded in the outgoing syslogs. The Class-of-Service policy-maps are enhanced with a new user-attribute *integer* keyword to store and indicate the threat-level.

You can configure the user-attribute *integer* at the [edit] class-of-service policy-map *policy-name* hierarchy.

The policy-map is referenced in each threat-level configuration to map the new *user-attribute*<> into the dfw filter term driving the configured action for each threat-level. The policy-map is used at the [edit] services web-filter profile *profile-name* security-intelligence-policy threat level *integer* policy-map *policy-name* hierarchy or [edit] services web-filter profile *profile-name* url-filter-template *template-name* security-intelligence-policy threat level *integer* policy-map *policy-name* hierarchy to map the threat level to a user-attribute.

For example,

```
[edit]
user@host set class-of-service policy-map threat1 user-attribute 1
user@host set class-of-service policy-map threat2 user-attribute 2
user@host set class-of-service policy-map threat3 user-attribute 3
...
...
user@host set class-of-service policy-map threat10 user-attribute 10
user@host set class-of-service policy-map white-list user-attribute 11
user@host set class-of-service policy-map black-list user-attribute 12
...
```

```
[edit]
user@host set services web-filter profile ATP-P1 security-intelligence-policy threat-level 1
threat-action log
```



```
user@host set services web-filter profile ATP-P1 security-intelligence-policy threat-level 1
policy-map threat1
```

SEE ALSO

[*security-intelligence-policy*](#)

[*security-intelligence*](#)

Configuring the Web Filter Profile for Sampling

IN THIS SECTION

- [Associate a Sampling Instance with the FPC | 437](#)
- [Configure a Sampling Instance and Associate the Template With the Sampling Instance. | 438](#)
- [Configure the sample instance and associate the flow-server IP address and other parameters. | 439](#)
- [Example: Configuring Web-filter Profile to Define Different Threat-Levels | 440](#)

Starting in Junos OS Release 19.3R1, web filtering process (url-filterd) supports inline sampling of packets as a threat level action. The packets are dropped, logged, and sampled based on the threat-action you configure. For scaled scenarios, sampling of packets is preferred over the logging option. Along with the existing threat level actions, you can configure the following threat level actions on the web filter profile at the edit web-filter profile *profile-name* security-intelligence-policy threat-level *threat-level* threat-action hierarchy level:

- drop-and-sample
- drop-log-and-sample
- log-and-sample
- sample

The inline flow monitoring samples the packets and sends the flow records in IPFIX format to a flow collector. You can derive the threat level for the sampled packets received at the external collector by matching the received IP from the sampled packets with the corresponding IP entry in `/var/db/url-filterd/urllf_si_cc_db.txt`. You can configure sampling using any of the following methods:

- Associate a sampling instance with the FPC on which the media interface is present at the [edit chassis] hierarchy level. If you are configuring sampling of IPv4 flows, IPv6 flows, or VPLS flows, you can configure the flow hash table size for each family.
- Configure the template properties for inline flow monitoring at the [edit services flow-monitoring hierarchy level.
- Configure a sampling instance and associate the flow-server IP address, port number, flow export rate, and specify the collectors at the [edit forwarding-options hierarchy level.

Associate a Sampling Instance with the FPC

To associate the defined instance with a particular FPC, MPC, or DPC, you include the `sampling-instance` statement at the [edit chassis fpc number] hierarchy level, as shown in the following example:

```
chassis {
  redundancy {
    graceful-switchover;
  }
  fpc 0 {
    pic0 {
      inline-services {
        bandwidth 10g;
      }
    }
  }
  pic 2 {
    inline-services {
      bandwidth 10g;
    }
  }
  pic 3 {
    inline-services {
      bandwidth 10g;
    }
  }
  sampling-instance 1to1;
  inline-services {
    flow-table-size {
      ipv4-flow-table-size 5;
      ipv6flow-table-size 5;
    }
  }
}
```

```

    }
}

```

Configure a Sampling Instance and Associate the Template With the Sampling Instance.

To configure the template properties for inline flow monitoring, include the following statements at the edit `services flow-monitoring` hierarchy level as shown in the following example:

```

services {
  flow-monitoring {
    version-ipfix {
      template ipv4 {
        flow-active-timeout 60;
      }
      flow-inactive-timeout 60;
      template-refresh-rate {
        packets 48000;
        seconds 60;
      }
      option-refresh-rate {
        packets 48000;
        seconds 60;
      }
      ipv4-template;
      template ipv6 {
        flow-active-timeout 60;
        flow-inactive-timeout 60;
        template-refresh-rate {
          packets 48000;
          seconds 60;
        }
        ipv6-template;
      }
    }
  }
}

```

Configure the sample instance and associate the flow-server IP address and other parameters.

To configure a sampling instance and associate the flow-server IP address and other parameters. include the following statements at the [edit forwarding-options] hierarchy, as shown in the following example:

```
forwarding-options {
  sampling {
    traceoptions {
      file ipfix.log size 10k;
    }
    instance {
      1to1 {
        input {
          rate 1;
        }
        family inet {
          output {
            flow-server 192.168.9.194;
            port 2055;;
            autonomous-system-type origin;
            version-ipfix {
              template {
                ipv4;
              }
            }
          }
        }
        inline-jflow {
          source-address 192.168.9.195;
        }
      }
    }
    family inet6 {
      output {
        flow-server 192.168.9.194;
        port 2000;
        autonomous-system-type origin;
        version-ipfix {
          template {
            ipv6;
          }
        }
      }
    }
  }
}
```

```

        inline-jflow {
            source-address 192.168.9.195;
        }
    }
}
}
}
}

```

Example: Configuring Web-filter Profile to Define Different Threat-Levels

```

web-filter {
    profile Profile1 ;
    security-intelligence-policy{
        file-type txt;
        threat-level 7 {
            threat-action {
                log-and-sample;
            }
        }
        threat-level 8 {
            threat-action {
                drop-log-and-sample;
            }
        }
        threat-level 10 {
            threat-action {
                drop-log-and-sample;
            }
        }
        threat-level 5{
            threat-action {
                drop-log-and-sample;
            }
        }
        threat-level 6 {
            threat-action {
                drop-log-and-sample;
            }
        }
        threat-level 9{
            threat-action {

```

```

        drop-log-and-sample;
    }
}
}
url-filter-template template1 {
    client-interfaces ge-0/0/4.0;
    client-routing-instance inet.0;
}
}
traceoptions {
    file webfilter_log size 1g;
    level all;
    flag all;
}
}
}

```

SEE ALSO

security-intelligence-policy

Configuring Traffic Sampling on MX, M and T Series Routers

GeoIP Filtering

IN THIS SECTION

- [Overview | 441](#)
- [How to Configure GeoIP Filtering on MX Series Routers | 442](#)

Overview

The GeoIP feeds are essentially a list of IP address to country code mappings. Starting in Junos OS 21.4R1, you can configure IP-based Geo locations on MX Series routers to fetch the GeoIP feeds from Policy Enforcer. By deploying the GeoIP feeds, you can enable the network to prevent devices from communicating with IP addresses belonging to specific countries.

You can configure the security intelligence process (IPFD) on MX series routers to fetch the GeoIP feeds from Policy Enforcer. Similar to existing C&C IP or IPv6 feeds, IPFD downloads the GeoIP feeds from the

Policy Enforcer. IPFD translates the feed in the file format that is processed by the web-filtering process (url-filterd) subsequently.

Starting in Junos OS 22.1R1, you can configure the security intelligence process (IPFD) on MX series routers to fetch the GeolP feeds from Juniper ATP Cloud. Similar to existing C&C IP or IPv6 feeds, IPFD downloads the GeolP feeds from the Juniper ATP Cloud.

How to Configure GeolP Filtering on MX Series Routers

The information fetched by the IPFD is saved in a file (**urlf_si_geoip_db.txt**) created at the **/var/db/url-filterd** location.

The format of the file sent by IPFD to the web filtering process is as follows:

IPv4 address\IPv6 address,Prefix,threat-level,VRF-name,Gen-num. Gen-num is always 0. *VRF-name* refers to a country code.

For example, 178.10.19.22,12,255,US,0

IPFD and the web-filtering process maintain a pconn connection for communicating the creation or update of files containing GeolP feeds. The Web-Filtering process enforces the GeolP feeds by programming the filters in the PFE to block the packets destined to the blocked countries. The APIs provided by liburlf are used to validate and parse the files.

The web-filtering process reads the file containing the list of IP addresses and the PFE filters are programmed with the destination IP addresses listed in the feed and the action configured for the associated country.

- **Global filter-** Countries are configured under global rule within a profile. All IP addresses for countries specific to that global rule are programmed in a single filter and applied to all templates in the profile. You can configure a profile to dynamically fetch GeolP feed by configuring geo-ip rule match country *country-name* at the [edit services web-filter profile *profile-name* security-intelligence-policy] hierarchy .
- **Group filter-** Groups of countries are configured under a template. All IP addresses associated with the countries for a Group are programmed in a group filter applied to the templates under which that group is configured. Group is a list of countries defined in a json file that is parsed by liburlf.

To configure a group filter, you must configure a json file at the **/var/db/url-filterd** location, where the **group.json** file contains the group mappings.

The format of the json file is as follows:

```
[
{
"group_name" : "group1",
```

```

"country" : ["ZA","YE"]
},
{
"group_name" : "group2",
"country" : ["YT"]
}
]

```

To dynamically fetch GeoIP feeds, you can configure a global filter using a single profile or configure multiple group filters using templates. We do not support both the configurations together.

The groups created in the json file are referred in the GeoIP match clause defined at the [edit services web-filter profile *profile-name* url-filter-template *template-name* security-intelligence-policy geo-ip rule match group *group-name*] hierarchy.

Global Allowlist and Global Blocklist

You can choose to customize the IP feed by adding your own allowlist and blocklist. This can be helpful to manage intelligence feeds that are custom to your security operations center or as a temporary measure for false positives. Starting in Junos OS release 21.4R1, you can allow or block certain IP addresses based on configuration through a CLI or a file. You can either configure separate list for allowlist and a separate list for blocklist or include the IP addresses in a file and include the file name in the CLI configuration.

You can create an IP-address-list at the [edit services web-filter] hierarchy. Here, IP-address-list contains the list of IP addresses that must be allowed or blocked. You can also create a file containing the IP addresses that need to be allowed or blocked in the **/var/db/url-filterd** location. The IP addresses configured as a part of the file or IP address list are programmed as a part of the global filter, which is attached to all templates.

You can define a global allowlist by configuring white-list (IP-address-list | *file-name*) at the edit services web-filter profile *profile-name* security-intelligence-policy hierarchy. You can define a global blocklist by configuring the black-list (IP-address-list | *file-name*) at the edit services web-filter profile *profile-name* security-intelligence-policy hierarchy. Here, the *IP-address-list*, refers to the name of IP address-list specified at the [edit services web-filter] hierarchy. The *file-name* refers to the name of the file which contains the list of the IP addresses that must be allowed or blocked. The file must be in the **/var/db/url-filterd** location and must have the same name as in the configuration.

The format of the global allowlist file is as follows:

Security Intelligence Policy Enforcement Version 2.0

```
IP Address,Prefix,Threat-level,VRF-Name,Gen-Num
198.51.100.1,32,0,junos-default-vrf,0
```

The format of the global blocklist file is as follows:

Security Intelligence Policy Enforcement Version 2.0

```
IP Address,Prefix,Threat-level,VRF-Name,Gen-Num
192.168.1.1,255,junos-default-vrf,0
```

The web-filtering process parses the list of global allowlist or global blocklist IP addresses and programs the implicit filter terms with the configured IP addresses to either allow or block the packets.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2 with the Next Gen Services as an Inline security capability
19.3R1	Starting in Junos OS Release 19.3R1, web filtering process (url-filterd) supports inline sampling of packets as a threat level action
18.4R1	Starting in Junos OS Release 18.4R1 with the Adaptive Services as an Inline security capability

9

PART

Aggregated Multiservices Interfaces

- [Enabling Load Balancing and High Availability Using Multiservices Interfaces | 446](#)
-

Enabling Load Balancing and High Availability Using Multiservices Interfaces

IN THIS CHAPTER

- [Understanding Aggregated Multiservices Interfaces for Next Gen Services | 446](#)
- [Configuring Aggregated Multiservices Interfaces | 452](#)
- [Configuring Load Balancing on AMS Infrastructure | 455](#)
- [Configuring Warm Standby for Services Interfaces | 459](#)

Understanding Aggregated Multiservices Interfaces for Next Gen Services

IN THIS SECTION

- [Aggregated Multiservices Interface | 446](#)
- [IPv6 Traffic on AMS Interfaces Overview | 449](#)
- [Member Failure Options and High Availability Settings | 450](#)
- [Warm Standby Redundancy | 451](#)

This topic provides an overview of using the Aggregated Multiservices Interfaces feature with the MX-SPC3 services card for Next Gen Services. It contains the following sections:

Aggregated Multiservices Interface

In Junos OS, you can combine multiple services interfaces to create a bundle of services interfaces that can function as a single interface. Such a bundle of interfaces is known as an *aggregated multiservices interface* (AMS), and is denoted as *amsN* in the configuration, where *N* is a unique number that identifies

an AMS interface (for example, `ams0`). Starting in Junos OS Release 19.3R2, AMS interfaces are supported on the Next Gen Services MX-SPC3 services card.

AMS configuration provides higher scalability, improved performance, and better failover and load-balancing options.

An AMS configuration enables service sets to support multiple services PICs by associating an AMS bundle with a service set. For Next Gen Services, the MX-SPC3 services card supports up to two PICs and you can have a maximum of eight MX-SPC3 services cards in your chassis. This enables a Next Gen Services AMS bundle to have up to 16 services PICs as member interfaces and you can distribute services among the member interfaces.

Member interfaces are identified as `mams` in the configuration. The `chassisd` process in routers that support AMS configuration creates a `mams` entry for every multiservices interface on the router.

When you configure services options at the `ams` interface level, the options apply to all member interfaces (`mams`) for the `ams` interface.

The options also apply to service sets configured on services interfaces corresponding to the `ams` interface's member interfaces. All settings are per PIC. For example, `session-limit` applies per member and not at an aggregate level.



NOTE: You cannot configure services options at both the `ams` (aggregate) and member-interface level. If services options are configured on `vms-x/y/z`, they also apply to service sets on `mams-x/y/z`.

When you want services options settings to apply uniformly to all members, configure services options at the `ams` interface level. If you need different settings for individual members, configure services options at the member interface level.



NOTE: Per-member drop of traffic and per-member next-hop configuration is required for NAT64. For NAPT44, this per-member specification allows arbitrary hash keys, providing better load-balancing options to allow dynamic NAT operations to be performed. For NAT64, NAPT44, and dynamic NAT44, it is not possible to determine which member allocates the dynamic NAT address. To ensure that reverse flow packets arrive at the same member as the forward flow packets, pool-address-based routes are used to steer reverse flow packets.



NOTE: If you modify a NAT pool that is being used by a service set assigned to an AMS interface, you must deactivate and activate the service set before the NAT pool changes take effect.

Traffic distribution over the member interfaces of an AMS interface can occur in either a round-robin fashion or hash-based. You can configure the following hash key values to regulate the traffic distribution: source-ip, destination-ip, and protocol. For services that require traffic symmetry, you must configure symmetrical hashing. Symmetrical hashing configuration ensures that both forward and reverse traffic is routed through the same member interface.

If the service set is applied on the Gigabit Ethernet or 10-Gigabit Ethernet interface (interface-style service set) that functions as the NAT inside interface, then the hash keys used for load balancing might be configured in such a way that the ingress key is set as destination IP address and the egress key is set as source IP address. Because the source IP address undergoes NAT processing, it is not available for hashing the traffic in the reverse direction. Therefore, load balancing does not happen on the same IP address and forward and reverse traffic does not map to the same PIC. With the hash keys reversed, load balancing occurs correctly.

With next-hop services, for forward traffic, the ingress key on the inside interface load-balances traffic, and for reverse traffic, the ingress key on the outside interface load-balances traffic or per-member next hops steer reverse traffic. With interface-style services, the ingress key load-balances forward traffic and the egress key load-balances forward traffic or per-member next hops steer reverse traffic. Forward traffic is traffic entering from the inner side of a service set and reverse traffic is traffic entering from the outer side of a service set. The forward key is the hash key used for the forward direction of traffic and the reverse key is the hash key used for the reverse direction of traffic (depends on whether it relates to interface services or next-hop services style.)

With stateful firewalls, you can configure the following combinations of forward and reverse keys for load balancing. In the following combinations presented for hash keys, FOR-KEY refers to the forward key, REV-KEY denotes the reverse key, SIP signifies source IP address, DIP signifies destination IP address, and PROTO refers to protocol such as IP.

- FOR-KEY: SIP, REV-KEY: DIP
- FOR-KEY: SIP,PROTO REV-KEY: DIP, PROTO
- FOR-KEY: DIP, REV-KEY: SIP
- FOR-KEY: DIP,PROTO REV-KEY: SIP, PROTO
- FOR-KEY: SIP,DIP REV-KEY: SIP, DIP
- FOR-KEY: SIP,DIP,PROTO REV-KEY: SIP, DIP,PROTO

With static NAT configured as basic NAT44 or destination NAT44, and with stateful firewall configured or not, if the forward direction of traffic must undergo NAT processing, configure the hash keys as follows:

- FOR-KEY: DIP, REV-KEY: SIP
- FOR-KEY: DIP,PROTO REV-KEY: SIP, PROTO

If the reverse direction of traffic must undergo NAT processing, configure the hash keys as follows:

- FOR-KEY: SIP, REV-KEY: DIP
- FOR-KEY: SIP,PROTO REV-KEY: DIP, PROTO

With dynamic NAT configured, and with stateful firewall configured or not, only the forward direction traffic can undergo NAT. The forward hash key can be any combination of SIP, DIP, and protocol, and the reverse hash key is ignored.



NOTE: The Junos OS AMS configuration supports IPv4 and IPv6 traffic.

IPv6 Traffic on AMS Interfaces Overview

You can use AMS interfaces for IPv6 traffic. To configure IPv6 support for an AMS interface, include the `family inet6` statement at the `[edit interfaces ams-interface-name unit 1]` hierarchy level. When `family inet` and `family inet6` are set for an AMS interface subunit, the hash-keys is configured at service-set level for interface style and at IFL level for next-hop style.

When a member interface of an AMS bundle fails, traffic destined to the failed member is redistributed among the remaining active members. The traffic (flows or sessions) traversing through the existing active members is unaffected. If M members are currently active, the expected result is that only about $1/M$ fraction of the traffic (flows/sessions) is impacted because that amount of traffic is shifted from the failed member to remain active members. When the failed member interface comes back online, only a fraction of the traffic is redistributed to the new member. If N members are currently active, the expected result is that only about $1/(N+1)$ fraction of the traffic (flows/sessions) is impacted because that amount of traffic moves to the new restored member. The $1/M$ and $1/(N+1)$ values assume that the flows are uniformly distributed among members, because a packet-hash is used to load-balance and because traffic usually contains a typical random combination of IP addresses (or any other fields that are used as load-balancing keys).

Similar to IPv4 traffic, for IPv6 packets, an AMS bundle must contain members of only one services PIC type.

The number of flows distributed, in an ideal environment, can be $1/N$ in a best-case scenario when the N th member goes up or down. However, this assumption considers that the hash keys load-balance the real or dynamic traffic. For example, consider a real-world deployment where member A is serving only one flow, whereas member B is serving 10 flows. If member B goes down, then the number of flows disrupted is $10/11$. The NAT pool-split behavior is designed to utilize the benefits of the rehash-minimization feature. The splitting of a NAT pool is performed for dynamic NAT scenarios (dynamic NAT, NAT64, and NAPT44).

If the original and redistributed flows are defined as follows:

- Member-original-flows—The traffic mapped to a member when all members are up.
- Member-redistributed-flows—The additional traffic mapped to a member when some other member fails. These traffic flows might need to be rebalanced when member interfaces come up and go down.

With the preceding definitions of the original and redistributed flows for member interfaces, the following observations apply:

- The member-original-flows of a member stay intact as long as that member is up. Such flows are not impacted when other members move between the up and down states.
- The member-redistributed-flows of a member can change when other members go up or down. This change of flows occurs because these additional flows need to be rebalanced among all active members. Therefore, the member-redistributed-flow can vary a lot based on other members going down or up. Although it might seem that when a member goes down, the flows on active-members are preserved, and that when a member goes up, flows on active-members are not preserved in an effective way, this behavior is only because of static or hash-based rebalancing of traffic among active members.

The rehash-minimization feature handles the operational changes in a member interface status only (such as member offline or member Junos OS reset). It does not handle changes in configuration. For example, addition or deletion, or activation and deactivation, of member interfaces at the `[edit interfaces ams/V load-balancing-options member-interface mams-a/b/0]` hierarchy level requires the member PICs to be bounced. Twice NAT or hairpinning is not supported, similar to IPv4 support for AMS interfaces.

Member Failure Options and High Availability Settings

Because multiple service interfaces are configured as part of an AMS bundle, AMS configuration also provides for failover and high availability support. You can either configure one of the member interfaces as a backup interface that becomes active when any one of the other member interfaces goes down, or configure the AMS in such a way that when one of the member interfaces goes down, the traffic assigned to that interface is shared across the active interfaces.

The `member-failure-options` *configuration statement* enables you to configure how to handle traffic when a member interface fails. One option is to redistribute the traffic immediately among the other member interfaces. However, redistribution of traffic involves recalculating the hash tags, and might cause some disruption in traffic on all the member interfaces.

The other option is to configure the AMS to drop all traffic that is assigned to the failed member interface. With this you can optionally configure an interval, `rejoin-timeout`, for the AMS to wait for the failed interface to come back online after which the AMS can redistribute the traffic among other member interfaces. If the failed member interface comes back online before the configured wait time, traffic continues unaffected on all member interfaces, including the interface that has come back online and resumed the operations.

You can also control the rejoining of the failed interface when it comes back online. If you do not include the `enable-rejoin` statement in the `member-failure-options` configuration, the failed interface cannot rejoin the AMS when it comes back online. In such cases, you can manually rejoin that to the AMS by executing the request `interfaces revert interface-name operational mode command`.

The `rejoin-timeout` and `enable-rejoin` statements enable you to minimize traffic disruptions when member interfaces flap.



NOTE: When `member-failure-options` are not configured, the default behavior is to drop member traffic with a rejoin timeout of 120 seconds.

The `high-availability-options` configuration enables you to designate one of the member interfaces as a backup interface. The backup interface does not participate in routing operations as long as it remains a backup interface. When a member interface fails, the backup interface handles the traffic assigned to the failed interface. When the failed interface comes back online, it becomes the new backup interface.

In a many-to-one configuration (N:1), a single backup interface supports all other member interfaces in the group. If any of the member interfaces fails, the backup interface takes over. In this stateless configuration, data is not synchronized between the backup interface and the other member interfaces.

When both `member-failure-options` and `high-availability-options` are configured for an AMS, the `high-availability-options` configuration takes precedence over the `member-failure-options` configuration. If a second failure occurs before the failed interface comes back online to be the new backup, the `member-failure-options` configuration takes effect.

Warm Standby Redundancy

Starting in Junos OS Release 19.3R2, the N:1 warm standby option is supported on the MX-SPC3 if you are running Next Gen Services. Each warm standby AMS interface contains two members; one member is the service interface you want to protect, called the primary interface, and one member is the secondary (backup) interface. The primary interface is the active interface and the backup interface does not handle any traffic unless the primary interface fails.

To configure warm standby on an AMS interface, you use the `redundancy-options` statement. You cannot use the `load-balancing-options` statement in a warm standby AMS interface.

To switch from the primary interface to the secondary interface, issue the request `interface switchover ams/N` command.

To revert to the primary interface from the secondary interface, issue the request `interface revert ams/N` command.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2, AMS interfaces are supported on the Next Gen Services MX-SPC3 services card.
19.3R2	Starting in Junos OS Release 19.3R2, the N:1 warm standby option is supported on the MX-SPC3 if you are running Next Gen Services.

Configuring Aggregated Multiservices Interfaces

The aggregated multiservices (AMS) interface configuration in Junos OS enables you to combine services interfaces from multiple PICs to create a bundle of interfaces that can function as a single interface. You identify the PIC that you want to act as the backup.

1. Create an aggregated multiservices interface and add member interfaces. Starting in Junos OS Release 19.3R2, an MX-SPC3 Next Gen Services AMS interface can have up to 14 member interfaces with a maximum of 7 MX-SPC3 services cards with up to 2 PICs on each card. Starting with Junos OS Release 16.2, an MS-MPC AMS interface can have up to 36 member interfaces. In Junos OS Release 16.1 and earlier, an AMS interface can have a maximum of 24 member interfaces.



NOTE: The member interface format is `mams-a/b/0`, where *a* is the Flexible PIC Concentrator (FPC) slot number and *b* is the PIC slot number.

```
[edit interfaces]
user@host# set interface-name load-balancing-options member-interface mams-a/b/0
user@host# set interface-name load-balancing-options member-interface mams-a/b/0
```

For example on an MS-MPC, which can have up to four PICs:

```
[edit interfaces]
user@host# set ams1 load-balancing-options member-interface mams-1/1/0
user@host# set ams1 load-balancing-options member-interface mams-1/2/0
```

For example on an MX-SPC3, which can have up to two PICs:

```
[edit interfaces]
user@host# set ams1 load-balancing-options member-interface mams-1/0/0
user@host# set ams1 load-balancing-options member-interface mams-1/1/0
```

2. Configure logical units for the AMS interface.

```
[edit interfaces]
user@host# set interface-name unit logical-unit-number family family
user@host# set interface-name unit logical-unit-number family family
```

For example:

```
[edit interfaces]
user@host# set ams1 unit 1 family inet
user@host# set ams1 unit 2 family inet6
```

3. Configure member failure options.

```
[edit interfaces interface-name]
user@host# set load-balancing-options member-failure-options drop-member-traffic rejoin-
timeout seconds
user@host# set load-balancing-options member-failure-options drop-member-traffic enable-rejoin
```

For example:

```
[edit interfaces ams1]
user@host# set load-balancing-options member-failure-options drop-member-traffic rejoin-
timeout 1000
user@host# set load-balancing-options member-failure-options drop-member-traffic enable-rejoin
```


4. Configure the preferred backup.

```
[edit interfaces interface-name]
user@host# set load-balancing-options high-availability-options many-to-one preferred-backup
preferred-backup
```


For example:

```
[edit interfaces ams1]
user@host# set load-balancing-options high-availability-options many-to-one preferred-backup
mams-1/2/0
```

5.

 **NOTE:** This step is not applicable to the Next Gen Services MX-SPC3 services card in the MX240, MX480 or MX960 chassis.

If the AMS interface has more than 24 member interfaces, set the service PIC boot timeout value to 240 or 300 seconds for every services PIC on the MX Series router. We recommend that you use a value of 240.

 **NOTE:** Starting with Junos OS Release 16.2, an AMS interface can have up to 36 member interfaces. In Junos OS Release 16.1 and earlier, an AMS interface could have a maximum of 24 member interfaces.

```
[edit interfaces interface-name multiservice-options]
user@host# set pic-boot-timeout (240 | 300);
```

For example:

```
[edit interfaces sp-1/1/0 multiservice-options]
user@host# set pic-boot-timeout 240
```

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2, an MX-SPC3 Next Gen Services AMS interface can have up to 16 member interfaces with a maximum of 8 MX-SPC3 services cards with up to 2 PICs on each card.
16.2	Starting with Junos OS Release 16.2, an MS-MPC AMS interface can have up to 36 member interfaces.

RELATED DOCUMENTATION

[Understanding Aggregated Multiservices Interfaces for Next Gen Services](#)

Configuring Load Balancing on AMS Infrastructure

IN THIS SECTION

- [Configuring AMS Infrastructure | 455](#)
- [Configuring High Availability | 457](#)
- [Load Balancing Network Address Translation Flows | 458](#)

Configuring load balancing requires an aggregated multiservices (AMS) system. AMS involves grouping several services PICs together. An AMS configuration eliminates the need for separate routers within a system. The primary benefit of having an AMS configuration is the ability to support load balancing of traffic across multiple services PICs.

AMS is supported on the MS-MPC and MS-MIC. Starting in Junos OS Release 19.3R2, AMS interfaces are supported on the MX-SPC3.

High availability (HA) is supported on AMS infrastructure on all MX Series 5G Universal Routing Platforms. AMS has several benefits:

- Support for configuring behavior if a services PIC that is part of the AMS configuration fails
- Support for specifying hash keys for each service set in either direction
- Support for adding routes to individual PICs within the AMS system

Configuring AMS Infrastructure

AMS supports load balancing across multiple service sets. All ingress or egress traffic for a service set can be load balanced across different services PICs. To enable load balancing, you have to configure an aggregate interface with existing services interfaces.

To configure failure behavior in AMS, include the `member-failure-options` statement:

```
[edit interfaces ams1]
load-balancing-options {
  member-failure-options {
    drop-member-traffic {
      rejoin-timeout rejoin-timeout;
    }
    redistribute-all-traffic {
      enable-rejoin;
    }
  }
}
```

If a PIC fails, you can configure the traffic to the failed PIC to be redistributed by using the `redistribute-all-traffic` statement at the `[edit interfaces interface-name load-balancing-options member-failure-options]` hierarchy level. If the `drop-member-traffic` statement is used, all traffic to the failed PIC is dropped. Both options are mutually exclusive.



NOTE: If `member-failure-options` is not explicitly configured, the default behavior is to drop member traffic with a rejoin timeout of 120 seconds.

Only `mams-` interfaces (services interfaces that are part of AMS) can be aggregated. After an AMS interface has been configured, you cannot configure the individual constituent `mams-` interfaces. A `mams-` interface cannot be used as an `ams` interface (this is not applicable to Next Gen Services MX-SPC3). AMS supports IPv4 (family `inet`) and IPv6 (family `inet6`). You cannot configure addresses on an AMS interface. Network Address Translation (NAT) is the only application that runs on AMS infrastructure at this time.



NOTE: You cannot configure unit 0 on an AMS interface.

To support multiple applications and different types of translation, AMS infrastructure supports configuring hashing for each service set. You can configure the hash keys separately for ingress and egress. The default configuration uses source IP, destination IP, and the protocol for hashing; incoming-interface for ingress and outgoing-interface for egress are also available.



NOTE: When using AMS in a load-balanced setup for the NAT solution, the number of NAT IP addresses must be greater than or equal to the number of active `mams-` interfaces you have added to the AMS bundle.

Configuring High Availability

In an AMS system configured with high availability, a designated services PIC acts as a backup for other active PICs that are part of the AMS system in a many-to-one (N:1) backup configuration. In a N:1 backup configuration, one PIC is available as backup for all other active PICs. If any of the active PICs fail, the backup PIC takes over for the failed PIC. In an N:1 (stateless) backup configuration, traffic states and data structures are not synchronized between the active PICs and the backup PIC.

An AMS system also supports a one-to-one (1:1) configuration. In the case of 1:1 backup, a backup interface is paired with a single active interface. If the active interface fails, the backup interface takes over. In a 1:1 (stateful) configuration, traffic states and data structures are synchronized between the active PICs and the backup PIC. Stateful synchronization is required for high availability of IPsec connections. For IPsec connections, AMS supports 1:1 configuration only.



NOTE: IPsec connections are not supported on the MX-SPC3 in this release.

High availability for load balancing is configured by adding the `high-availability-options` statement at the `[edit interfaces interface-name load-balancing-options]` hierarchy level.

To configure N:1 high availability, include the `high-availability-options` statement with the `many-to-one` option:

```
[edit interfaces ams1]
load-balancing-options {
  high-availability-options {
    many-to-one {
      preferred-backup preferred-backup;
    }
  }
}
```

Starting in Junos OS Release 16.1, you can configure stateful 1:1 high availability on an MS-MPC. To configure stateful 1:1 high availability, at the `[edit interfaces interface-name load-balancing-options]` hierarchy level, include the `high-availability-options` statement with the `one-to-one` option:



NOTE: The Next Gen Services MX-SPC3 services card does not support AMS 1:1 high availability.

```
[edit interfaces ams1]
load-balancing-options {
```

```

high-availability-options {
    one-to-one {
        preferred-backup preferred-backup;
    }
}

```

Load Balancing Network Address Translation Flows

Network Address Translation (NAT) has been programmed as a plug-in and is a function of load balancing and high availability. The plug-in runs on AMS infrastructure. All flows for translation are automatically distributed to different services PICs that are part of the AMS infrastructure. In case of failure of an active services PIC, the configured backup PIC takes over the NAT pool resources of the failed PIC. The hashing method selected depends on the type of NAT. Using NAT on AMS infrastructure has a few limitations:

- NAT flows to failed PICs cannot be restored.
- There is no support for IPv6 flows.

IPv6 address pools are not supported with AMS, however NAT64 is supported with AMS, so that IPv6 flows enters AMS.

NAT64 is supported for Next Gen Services on the MX-SPC3 services card, there is no support of NAT66. IPv6 flows for different NAT services are supported except where the translation is required to be IPv6 to IPv6 or IPv4 to IPv6.

- Twice NAT is not supported for load balancing on MS-MPC cards.

Twice NAT is supported for load balancing on the Next Gen Services MX-SPC3 services card.

- Deterministic NAT uses warm-standby AMS configuration and can distribute the load using multiple AMS bundles in warm-standby mode.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2, AMS interfaces are supported with the MX-SPC3.
16.1	Starting in Junos OS Release 16.1, you can configure stateful 1:1 high availability on an MS-MPC.

Configuring Warm Standby for Services Interfaces

You can configure an N:1 warm standby option for MS-MPCs, MS-MICs, and MX-SPC3s by creating multiple aggregated multiservices (AMS) interfaces, each of which contains the service interface you want to backup and the service interface that acts as the backup. The same backup service interface can be used in all these AMS interfaces. Starting in Junos OS Release 19.3R2, the N:1 warm standby option is supported on the MX-SPC3.

To configure warm standby for services interfaces:

1. Create an AMS interface.

```
[edit interfaces]
user@host# set amsN
```

The variable *N* is a unique number, such as 0 or 1.

2. Specify the primary service interface that you want to backup.

```
[edit interfaces amsN]
user@host# set redundancy-options primary mams-a/b/0
```

The variable *a* is the FPC slot number and *b* is the PIC slot number for the primary service interface.

3. Specify the secondary service interface, which backs up the primary interface.

```
[edit interfaces amsN]
user@host# set redundancy-options secondary mams-a/b/0
```

The variable *a* is the FPC slot number and *b* is the PIC slot number for the secondary service interface.

4. Repeat Steps 1 through 3 to create an AMS interface for each service interface that you want to backup. You can use the same secondary service interface in each AMS interface.

Change History Table

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
19.3R2	Starting in Junos OS Release 19.3R2, the N:1 warm standby option is also supported on the MX-SPC3 if you are running Next Gen Services.

RELATED DOCUMENTATION

| *Understanding Aggregated Multiservices Interfaces*

10

PART

Inter-Chassis Services PIC High Availability

-
- [Inter-Chassis Services PIC High Availability Overview and Configuration | 462](#)
-

Inter-Chassis Services PIC High Availability Overview and Configuration

IN THIS CHAPTER

- [Next Gen Services Inter-chassis High Availability Overview for NAT, Stateful Firewall, and IDS Flows | 462](#)
- [Inter-Chassis Stateful Synchronization for Long Lived NAT, Stateful Firewall, and IDS Flows for Next Gen Services | 476](#)
- [Inter-Chassis Services Redundancy Overview for Next Gen Services | 485](#)
- [Configuring Inter-Chassis Services Redundancy for Next Gen Services | 488](#)

Next Gen Services Inter-chassis High Availability Overview for NAT, Stateful Firewall, and IDS Flows

IN THIS SECTION

- [Inter-chassis High Availability Overview for NAT, Stateful Firewall, and IDS Flows for Next Gen Services | 463](#)
- [Example: Next Gen Services Inter-Chassis Stateful High Availability for NAT and Stateful Firewall \(MX-SPC3\) | 463](#)

Inter-chassis High Availability Overview for NAT, Stateful Firewall, and IDS Flows for Next Gen Services

IN THIS SECTION

- [Benefits](#) | 463

Carrier-grade NAT, stateful firewall, and IDS flows can be configured with a dual-chassis, redundant data path. Although intra-chassis high availability can be used in an MX Series device by employing the AMS interfaces, this method only deals locally with services PIC failures. If for any reason traffic is switched to a backup router due to some other failure in the router, the session state from the services PIC is lost unless you configure synchronization of the services session states with a services PIC on the backup router.

Inter-chassis high availability provides this synchronization, and controls switchovers between the services PICs in the redundancy pair. Inter-chassis high availability is a primary-secondary model, not an active-active cluster. Only one services PIC in a redundancy pair, the current primary, receives traffic to be serviced.

To configure interchassis high availability for NAT, stateful firewall, and IDS, you configure:

1. Stateful synchronization, which replicates the session state from the primary services PICs on the primary to the backup services PIC on the other chassis.
2. Inter-chassis services redundancy, which controls primary role switchovers in the services PIC redundancy pair, based on monitored events. Most operators would not want to employ stateful synchronization without also implementing services redundancy.

Benefits

Interchassis high availability provides automatic switchovers from a services PIC on one chassis to a services PIC on another chassis, while providing uninterrupted services for customer traffic.

Example: Next Gen Services Inter-Chassis Stateful High Availability for NAT and Stateful Firewall (MX-SPC3)

IN THIS SECTION

- [Requirements](#) | 464

- [Overview | 464](#)
- [Configuration | 464](#)

This example shows how to configure Next Gen Services inter-chassis high availability for stateful firewall and NAT services.

Requirements

This example uses the following hardware and software components:

- Two MX480 routers with MX-SPC3 services cards
- Junos OS Release 19.3R2, 19.4R1 or later

Overview

Two MX 3D routers are identically configured to facilitate stateful failover for firewall and NAT services in case of a chassis failure.

Configuration

IN THIS SECTION

- [CLI Quick Configuration | 465](#)
- [Configuring Interfaces for Chassis 1. | 467](#)
- [Configure Routing Information for Chassis 1 | 469](#)
- [Configuring NAT and Stateful Firewall for Chassis 1 | 470](#)
- [Configuring the Service Set | 472](#)
- [Configuring Interfaces for Chassis 2 | 473](#)
- [Configure Routing Information for Chassis 2 | 475](#)

To configure inter-chassis high availability for this example, perform these tasks:

CLI Quick Configuration

To quickly configure this example on the routers, copy the following commands and paste them into the router terminal window after removing line breaks and substituting interface information specific to your site.



NOTE: The following configuration is for chassis 1.

```
[edit]
set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.2
set interfaces vms-4/0/0 redundancy-options routing-instance HA
set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.1/32
set interfaces vms-4/0/0 unit 20 family inet
set interfaces vms-4/0/0 unit 20 service-domain inside
set interfaces vms-4/0/0 unit 30 family inet
set interfaces vms-4/0/0 unit 30 service-domain outside
set interfaces ge-2/0/0 vlan-tagging
set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.1/24
set routing-instances HA instance-type vrf
set routing-instances HA interface ge-2/0/0.0
set routing-instances HA interface vms-4/0/0.10
set routing-instances HA route-distinguisher 1:1
set policy-options policy-statement dummy term 1 then reject
set routing-instances HA vrf-import dummy
set routing-instances HA vrf-export dummy
set routing-instances HA routing-options static route route 5.5.5.1/32 next-hop vms-4/0/0.10
set routing-instances HA routing-options static route route 5.5.5.2/32 next-hop 20.1.1.2
set services nat pool p2 address 32.0.0.0/24
set services nat pool p2 port automatic random-allocation
set services nat pool p2 address-allocation round-robin
set services nat rule r2 match-direction input
set services nat rule r2 term t1 from source-address 129.0.0.0/8
set services nat rule r2 term t1 from source-address 128.0.0.0/8
set services nat rule r2 term t1 then translated source-pool p2
set services nat rule r2 term t1 then translated translation-type napt-44
set services nat rule r2 term t1 then translated address-pooling paired
set services nat rule r2 term t1 then syslog
set services stateful-firewall rule r2 match-direction input
set services stateful-firewall rule r2 term t1 from source-address any-unicast
set services stateful-firewall rule r2 term t1 then accept
```

```

set services stateful-firewall rule r2 term t1 then syslog
set services service-set ss2 replicate-services replication-threshold 180
set services service-set ss2 replicate-services stateful-firewall
set services service-set ss2 replicate-services nat
set services service-set ss2 stateful-firewall-rules r2
set services service-set ss2 nat-rules r2
set services service-set ss2 next-hop-service inside-service-interface vms-4/0/0.20
set services service-set ss2 next-hop-service outside-service-interface vms-4/0/0.30
set services service-set ss2 syslog host local class session-logs
set services service-set ss2 syslog host local class stateful-firewall-logs
set services service-set ss2 syslog host local class nat-logs

```



NOTE: The following configuration is for chassis 2. The NAT, stateful firewall, and service-set information must be identical for chassis 1 and 2.

```

set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.1
set interfaces vms-4/0/0 redundancy-options routing-instance HA
set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.2/32
set interfaces vms-4/0/0 unit 20 family inet
set interfaces vms-4/0/0 unit 20 service-domain inside
set interfaces vms-4/0/0 unit 30 family inet
set interfaces vms-4/0/0 unit 30 service-domain outside
set interfaces ge-2/0/0 vlan-tagging
set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.2/24
set routing-instances HA instance-type vrf
set routing-instances HA interface ge-2/0/0.0
set routing-instances HA interface vms-4/0/0.10
set routing-instances HA route-distinguisher 1:1
set policy-options policy-statement dummy term 1 then reject
set routing-instances HA vrf-import dummy
set routing-instances HA vrf-export dummy
set routing-instances HA routing-options static route 5.5.5.2/32 next-hop vms-4/0/0.10
set routing-instances HA routing-options static route 5.5.5.1/32 next-hop 20.1.1.1
set services nat pool p2 address 32.0.0.0/24
set services nat pool p2 port automatic random-allocation
set services nat pool p2 address-allocation round-robin
set services nat rule r2 match-direction input
set services nat rule r2 term t1 from source-address 129.0.0.0/8
set services nat rule r2 term t1 from source-address 128.0.0.0/8

```

```

set services nat rule r2 term t1 then translated source-pool p2
set services nat rule r2 term t1 then translated translation-type napt-44
set services nat rule r2 term t1 then translated address-pooling paired
set services nat rule r2 term t1 then syslog
set services stateful-firewall rule r2 match-direction input
set services stateful-firewall rule r2 term t1 from source-address any-unicast
set services stateful-firewall rule r2 term t1 then accept
set services stateful-firewall rule r2 term t1 then syslog
set services service-set ss2 replicate-services replication-threshold 180
set services service-set ss2 replicate-services stateful-firewall
set services service-set ss2 replicate-services nat
set services service-set ss2 stateful-firewall-rules r2
set services service-set ss2 nat-rules r2
set services service-set ss2 next-hop-service inside-service-interface vms-4/0/0.20
set services service-set ss2 next-hop-service outside-service-interface vms-4/0/0.30
set services service-set ss2 syslog host local class session-logs
set services service-set ss2 syslog host local class stateful-firewall-logs
set services service-set ss2 syslog host local class nat-logs

```

Configuring Interfaces for Chassis 1.

Step-by-Step Procedure

The interfaces for each of the HA pair of routers are configured identically with the exception of the following service PIC options:

- `redundancy-options redundancy-peer ipaddress address`
- `unit unit-number family inet address address` of a unit, other than 0, that contains the `ip-address-owner` service-plane option

To configure interfaces:

1. Configure the redundant service PIC on chassis 1.

```

[edit interfaces]
user@host# set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.2
user@host# set interfaces vms-4/0/0 redundancy-options routing-instance HA
user@host# set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
user@host# set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.1/32
user@host# set interfaces vms-4/0/0 unit 20 family inet
user@host# set interfaces vms-4/0/0 unit 20 service-domain inside

```



```

user@host# set interfaces vms-4/0/0 unit 30 family inet
user@host# set interfaces vms-4/0/0 unit 30 service-domain outside

```

2. Configure the interfaces for chassis 1 that are used as interchassis links for synchronization traffic.

```

user@host# set interfaces ge-2/0/0 vlan-tagging
user@host# set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.1/24

```

3. Configure remaining interfaces as needed.

Results

```

user@host# show interfaces
ge-2/0/0 {
    vlan-tagging;
    unit 0 {
        vlan-id 100;
        family inet {
            address 20.1.1.1/24;
        }
    }
}
vms-4/0/0 {
    redundancy-options {
        redundancy-peer {
            ipaddress 5.5.5.2;
        }
        routing-instance HA;
    }
    unit 10 {
        ip-address-owner service-plane;
        family inet {
            address 5.5.5.1/32;
        }
    }
    unit 20 {
        family inet;
        family inet6;
        service-domain inside;
    }
}

```

```

        unit 30 {
            family inet;
            family inet6;
            service-domain outside;
        }
    }
}

```

Configure Routing Information for Chassis 1

Step-by-Step Procedure

Detailed routing configuration is not included for this example. A routing instance is required for the HA synchronization traffic between the chassis as follows:

- Configure routing instances for Chassis 1.

```

user@host# set routing-instances HA instance-type vrf
user@host# set routing-instances HA interface ge-2/0/0.0
user@host# set routing-instances HA interface vms-4/0/0.10
user@host# set routing-instances HA route-distinguisher 1:1
user@host# set policy-options policy-statement dummy term 1 then reject
user@host# set routing-instances HA vrf-import dummy
user@host# set routing-instances HA vrf-export dummy
user@host# set routing-instances HA routing-options static route route 5.5.5.1/32 next-hop
vms-4/0/0.10
user@host# set routing-instances HA routing-options static route route 5.5.5.2/32 next-hop
20.1.1.2

```

Results

```

user@host# show routing-instances
HA {
    instance-type vrf;
    interface ge-2/0/0.0;
    interface vms-4/0/0.10;
    route-distinguisher 1:1;
    vrf-import dummy;
    vrf-export dummy;
    routing-options {

```

```

        static {
            route 5.5.5.1/32 next-hop vms-4/0/0.10;
            route 5.5.5.2/32 next-hop 20.1.1.2;
        }
    }
}

```

Configuring NAT and Stateful Firewall for Chassis 1

Step-by-Step Procedure

Configure NAT and stateful firewall identically on both routers. To configure NAT and stateful firewall:

1. Configure NAT as needed.

```

user@host# set services nat pool p2 address 32.0.0.0/24
user@host# set services nat pool p2 port automatic random-allocation
user@host# set services nat pool p2 address-allocation round-robin
user@host# set services nat rule r2 match-direction input
user@host# set services nat rule r2 term t1 from source-address 129.0.0.0/8
user@host# set services nat rule r2 term t1 from source-address 128.0.0.0/8
user@host# set services nat rule r2 term t1 then translated source-pool p2
user@host# set services nat rule r2 term t1 then translated translation-type napt-44
user@host# set services nat rule r2 term t1 then translated address-pooling paired
user@host# set services nat rule r2 term t1 then syslog

```

2. Configure stateful firewall as needed.

```

user@host# set services stateful-firewall rule r2 match-direction input
user@host# set services stateful-firewall rule r2 term t1 from source-address any-unicast
user@host# set services stateful-firewall rule r2 term t1 then accept
user@host# set services stateful-firewall rule r2 term t1 then syslog

```

Results

```

user@host# show services nat
nat {
    pool p2 {
        address 32.0.0.0/24;
    }
}

```

```

        port {
            automatic {
                random-allocation;
            }
        }
        address-allocation round-robin;
    }
    rule r2 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    129.0.0.0/8;
                    128.0.0.0/8;
                }
            }
            then {
                translated {
                    source-pool p2;
                    translation-type {
                        napt-44;
                    }
                    address-pooling paired;
                }
                syslog;
            }
        }
    }
}

```

user@host **show services stateful-firewall**

```

rule r2 {
    match-direction input;
    term t1 {
        from {
            source-address {
                any-unicast;
            }
        }
        then {

```

```

        accept;
        syslog;
    }
}
}

```

Configuring the Service Set

Step-by-Step Procedure

Configure the the service set identically on both routers. To configure the service set:

1. Configure the service set replication options.

```

user@host# set services service-set ss2 replicate-services replication-threshold 180
user@host# set services service-set ss2 replicate-services stateful-firewall
user@host# set services service-set ss2 replicate-services nat

```

2. Configure references to NAT and stateful firewall rules for the service set.

```

user@host# set services service-set ss2 stateful-firewall-rules r2
user@host# set services service-set ss2 nat-rules r2

```

3. Configure next-hop service interface on the vms-PIC.

```

user@host# set services service-set ss2 next-hop-service inside-service-interface vms-4/0/0.20
user@host# set services service-set ss2 next-hop-service outside-service-interface
vms-4/0/0.30

```

4. Configure desired logging options.

```

user@host# set services service-set ss2 syslog host local class session-logs
user@host# set services service-set ss2 syslog host local class stateful-firewall-logs
user@host# set services service-set ss2 syslog host local class nat-logs

```

Results

```

user@host# show services service-set ss2
syslog {
    host local {
        class {
            session-logs;
            inactive: stateful-firewall-logs;
            nat-logs;
        }
    }
}
replicate-services {
    replication-threshold 180;
    stateful-firewall;
    nat;
}
stateful-firewall-rules r2;
inactive: nat-rules r2;
next-hop-service {
    inside-service-interface vms-3/0/0.20;
    outside-service-interface vms-3/0/0.30;
}
}

```

Configuring Interfaces for Chassis 2

Step-by-Step Procedure

The interfaces for each of the HA pair of routers are configured identically with the exception of the following service PIC options:

- `redundancy-options redundancy-peer ipaddress address`
- `unit unit-number family inet address address` of a unit, other than 0, that contains the `ip-address-owner service-plane option`

1. Configure the redundant service PIC on chassis 2.

The redundancy-peer ipaddress points to the address of the unit (unit 10) on vms-4/0/0 on chassis on chassis 1 that contains the ip-address-owner service-plane statement.

```
[edit interfaces]
set interfaces vms-4/0/0 redundancy-options redundancy-peer ipaddress 5.5.5.1
user@host# set interfaces vms-4/0/0 redundancy-options routing-instance HA
user@host# set interfaces vms-4/0/0 unit 10 ip-address-owner service-plane
user@host# set interfaces vms-4/0/0 unit 10 family inet address 5.5.5.2/32
user@host# set interfaces vms-4/0/0 unit 20 family inet
user@host# set interfaces vms-4/0/0 unit 20 service-domain inside
user@host# set interfaces vms-4/0/0 unit 30 family inet
user@host# set interfaces vms-4/0/0 unit 30 service-domain outside
```

2. Configure the interfaces for chassis 2 that are used as interchassis links for synchronization traffic

```
user@host# set interfaces ge-2/0/0 vlan-tagging
user@host# set interfaces ge-2/0/0 unit 0 vlan-id 100 family inet address 20.1.1.2/24
```

3. Configure remaining interfaces for chassis 2 as needed.

Results

```
user@host# show interfaces
vms-4/0/0 {
    redundancy-options {
        redundancy-peer {
            ipaddress 5.5.5.1;
        }
        routing-instance HA;
    }
    unit 0 {
        family inet;
    }
    unit 10 {
        ip-address-owner service-plane;
        family inet {
            address 5.5.5.2/32;
        }
    }
}
ge-2/0/0 {
```

```

vlan-tagging;
unit 0 {
    vlan-id 100;
    family inet {
        address 20.1.1.2/24;
    }
}
unit 10 {
    vlan-id 10;
    family inet {
        address 2.10.1.2/24;
    }
}

```

Configure Routing Information for Chassis 2

Step-by-Step Procedure

Detailed routing configuration is not included for this example. A routing instance is required for the HA synchronization traffic between the two chassis and is included here.

- Configure routing instances for chassis 2.

```

user@host# set routing-instances HA instance-type vrf
user@host# set routing-instances HA interface ge-2/0/0.0
user@host# set routing-instances HA interface vms-4/0/0.10
user@host# set routing-instances HA route-distinguisher 1:1
user@host# set policy-options policy-statement dummy term 1 then reject
user@host# set routing-instances HA vrf-import dummy
user@host# set routing-instances HA vrf-export dummy
user@host# set routing-instances HA routing-options static route 5.5.5.2/32 next-hop
vms-4/0/0.10
user@host# set routing-instances HA routing-options static route 5.5.5.1/32 next-hop 20.1.1.1

```



NOTE: The following configuration steps are *identical* to the steps shown for chassis 1.

- Configuring NAT and Stateful Firewall
- Configuring the Service Set

Results

```
user@host# show services routing-instances
HA {
    instance-type vrf;
    interface xe-2/2/0.0;
    interface vms-4/0/0.10;
    route-distinguisher 1:1;
    vrf-import dummy;
    vrf-export dummy;
    routing-options {
        static {
            route 5.5.5.2/32 next-hop vms-4/0/0.10;
            route 5.5.5.1/32 next-hop 20.1.1.1;
        }
    }
}
```

RELATED DOCUMENTATION

[Inter-Chassis Stateful Synchronization for Long Lived NAT, Stateful Firewall, and IDS Flows for Next Gen Services | 476](#)

[Inter-Chassis Services Redundancy Overview for Next Gen Services | 485](#)

Inter-Chassis Stateful Synchronization for Long Lived NAT, Stateful Firewall, and IDS Flows for Next Gen Services

IN THIS SECTION

- [Inter-Chassis Stateful Synchronization Overview | 477](#)
- [Configuring Inter-Chassis Stateful Synchronization for Long- Lived NAT, Stateful Firewall, and IDS Flows for Next Gen Services | 478](#)

Inter-Chassis Stateful Synchronization Overview

IN THIS SECTION

- [Benefits | 478](#)

Stateful synchronization replicates the state of long-lived NAT, stateful firewall, and IDS sessions on the primary services PIC and sends it to the backup services PIC, which is on a different MX Series chassis. By default, long lived sessions are defined as having been active on the services PIC for at least 180 seconds, though you can configure this to a higher value.

The following restrictions apply:

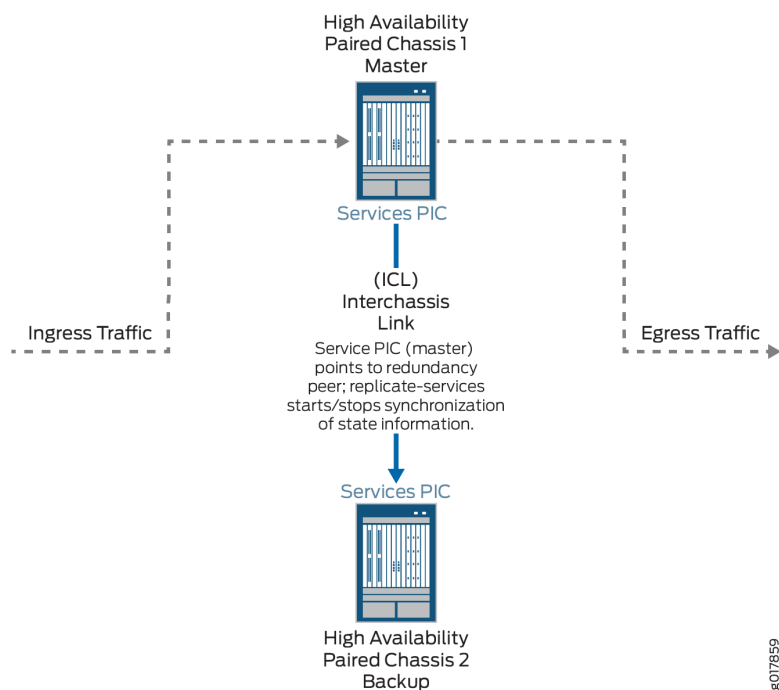
- NAPT44 is the only translation type supported.

Replicating state information for the port block allocation (PBA), endpoint-independent mapping (EIM), or endpoint-independent filters (EIF) features are supported supported for Next Gen Services.

When configuring a service set for NAT, stateful firewall, or IDS that belongs to a stateful synchronization setup, you must use a next-hop service set, and the NAT, stateful firewall, and IDS configurations for the service set must be identical on both MX Series chassis.

[Figure 15 on page 478](#) shows the stateful synchronization topology.

Figure 15: Stateful Sync Topology



Benefits

Interchassis stateful synchronization of the services session state allows uninterrupted services when a switchover occurs from a services PIC on one chassis to a services PIC on another chassis.

Configuring Inter-Chassis Stateful Synchronization for Long- Lived NAT, Stateful Firewall, and IDS Flows for Next Gen Services

IN THIS SECTION

- [Configuring Inter-Chassis Stateful Synchronization for Next Gen Services with non-AMS Interface | 479](#)
- [Configuring Inter-Chassis Stateful Synchronization for Next Gen Services with AMS Interface | 481](#)

Configuring Inter-Chassis Stateful Synchronization for Next Gen Services with non-AMS Interface

To configure stateful synchronization inter-chassis high availability for NAT, stateful firewall, and IDS flows for Next Gen Services when the services interfaces are not AMS, perform the following configuration steps on each chassis of the high availability pair.

1. Specify the IP address of the vms- interface. This address is used by the TCP channel between the HA pairs.

```
[edit interfaces interface-name redundancy-options]
user@host# set redundancy-local data-address address
```

For example:

```
[edit interfaces vms-1/0/0 redundancy-options]
user@host# set redundancy-local data-address 192.0.2.2
```

When you configure the other chassis, this is the address you use for the redundancy-peer ipaddress.

2. Specify the IP address of the remote services interface. This address is used by the TCP channel between the HA pairs.

```
[edit interfaces interface-name redundancy-options]
user@host# set redundancy-peer ipaddress address
```

For example:

```
[edit interfaces vms-1/0/0 redundancy-options]
user@host# set redundancy-peer ipaddress 192.0.2.1
```

When you configure the other chassis, this is the address you use for the redundancy-local data-address.

3. Configure the length of time that the flow remains active for replication, in seconds.

```
[edit interfaces interface-name redundancy-options]
user@host# set replication-threshold seconds
```

For example:

```
[edit interfaces vms-1/0/0 redundancy-options]
user@host# set replication-threshold 60
```

4. Configure a unit other than 0, and assign it the IP address of the local services interface that you configured with the `redundancy-local data-address` option.

```
[edit interfaces interface-name]
user@host# set unit logical-unit-number family (inet | inet6) address address
```

For example:

```
[edit interfaces vms-1/0/0]
user@host# set unit 10 family inet address 192.0.2.2/32
```

5. For ease of management, we recommend you create a special routing instance with `instance-type vrf` to host the HA synchronization traffic between the MX Series high availability pair. Then specify the name of the special routing instance to apply to the HA synchronization traffic between the high availability pair.

```
[edit interfaces interface-name redundancy-options]
user@host# set routing-instance instance-name
```

6. Configure the inside and outside interface units, which are used by the next-hop service set. Use different unit numbers for the inside and outside units, and do not use 0 or the unit number used in Step 4.

```
[edit]
user@host# set interfaces interface-name unit logical-unit-number family (inet | inet6)
user@host# set interfaces interface-name unit logical-unit-number service-domain inside
user@host# set interfaces interface-name unit logical-unit-number family (inet | inet6)
user@host# set interfaces interface-name unit logical-unit-number service-domain outside
```

For example:

```
[edit]
user@host# set interfaces vms-1/0/0 unit 100 family inet
user@host# set interfaces vms-1/0/0 unit 100 family inet6
```

```

user@host# set interfaces vms-1/0/0 unit 100 service-domain inside
user@host# set interfaces vms-1/0/0 unit 1000 family inet
user@host# set interfaces vms-1/0/0 unit 1000 family inet6
user@host# set interfaces vms-1/0/0 unit 1000 service-domain outside

```

7. Configure the next-hop service set that contains the NAT rules, stateful firewall rules, or IDS screens. The service set must be configured identically on each chassis of the high availability pair. The NAT rules, stateful firewall rules, and IDS screens must also be configured identically on each chassis.

For example:

```

user@host#set service-set internal-nat next-hop-service inside-service-interface vms-1/0/0.100
user@host#set service-set internal-nat next-hop-service outside-service-interface
vms-1/0/0.1000
user@host#set service-set internal-nat next-hop-service nat-rules internal-nat1

```

8. Repeat these steps for the other chassis of the high availability pair.

Configuring Inter-Chassis Stateful Synchronization for Next Gen Services with AMS Interface

To configure stateful synchronization inter-chassis high availability for NAT, stateful firewall, and IDS flows for Next Gen Services for an AMS services interface, perform the following configuration steps on each chassis of the high availability pair.

1. Configure a services vms- interface for every member of the AMS interface:
 - a. Specify the IP address of the vms- interface. This address is used by the TCP channel between the HA pairs.

```

[edit interfaces interface-name redundancy-options]
user@host# set redundancy-local data-address address

```

For example:

```

[edit interfaces vms-1/0/0 redundancy-options]
user@host# set redundancy-local data-address 192.0.2.2

```

When you configure the other chassis, this is the address you use for the redundancy-peer ipaddress.

- b. Specify the IP address of the remote services interface. This address is used by the TCP channel between the HA pairs.

```
[edit interfaces interface-name redundancy-options]
user@host# set redundancy-peer ipaddress address
```

For example:

```
[edit interfaces vms-1/0/0 redundancy-options]
user@host# set redundancy-peer ipaddress 192.0.2.1
```

When you configure the other chassis, this is the address you use for the redundancy-local data-address.

- c. Configure the length of time that the flow remains active for replication, in seconds.

```
[edit interfaces interface-name redundancy-options]
user@host# set replication-threshold seconds
```

For example:

```
[edit interfaces vms-1/0/0 redundancy-options]
user@host# set replication-threshold 60
```

- d. Configure a unit other than 0, and assign it the IP address of the local services interface that you configured with the redundancy-local data-address option.

```
[edit interfaces interface-name]
user@host# set unit logical-unit-number family inet address address
```

For example:

```
[edit interfaces vms-1/0/0]
user@host# set unit 10 family inet address 192.0.2.2/32
```

- e. For ease of management, we recommend you create a special routing instance with instance-type vrf to host the HA synchronization traffic between the MX Series high availability pair. Then

specify the name of the special routing instance to apply to the HA synchronization traffic between the high availability pair.

```
[edit interfaces interface-name redundancy-options]
user@host# set routing-instance instance-name
```

2. Create the AMS interface and add the member interfaces you configured in Step 1.

```
[edit interfaces]
user@host# set interface-name load-balancing-options [member-interface mams-a/b/0]
```

where the *interface-name* is *amsN*, and *a* is the FPC slot number and *b* is the PIC slot number for each member interface.

For example:

```
[edit interfaces]
user@host# set ams0 load-balancing-options member-interface mams-1/0/0
user@host# set ams0 load-balancing-options member-interface mams-1/1/0
```

3. Configure the inside interface for the AMS interface, which is used by the next-hop service set:
 - a. Configure the family for the inside interface. Do not use 0 for the unit number.

```
[edit]
user@host# set interfaces interface-name unit logical-unit-number service-domain inside
user@host# set interfaces interface-name unit logical-unit-number family (inet | inet6)
```

For example:

```
[edit]
user@host# set interfaces ams0 unit 100 service-domain inside
user@host# set interfaces ams0 unit 100 family inet
user@host# set interfaces ams0 unit 100 family inet6
```

- b. Configure the hash key to regulate distribution for the inside interface.

```
[edit set interfaces interface-name unit logical-unit-number]
user@host# load-balancing-options hash-keys ingress-key [source-ip destination-ip]
```


4. Configure the outside interface for the AMS interface, which is used by the next-hop service set. Do not use 0 or the same unit number that you used for the inside interface.
 - a. Configure the family for the outside interface.

```
[edit]
user@host# set interfaces interface-name unit logical-unit-number service-domain outside
user@host# set interfaces interface-name unit logical-unit-number family (inet | inet6)
```

For example:

```
[edit]
user@host# set interfaces ams0 unit 1000 service-domain outside
user@host# set interfaces ams0 unit 1000 family inet
user@host# set interfaces ams0 unit 1000 family inet6
```

- b. Configure the hash key to regulate distribution for the outside interface.

```
[edit set interfaces interface-name unit logical-unit-number]
user@host# load-balancing-options hash-keys ingress-key [source-ip destination-ip]
```

5. Configure the next-hop service set that contains the NAT rules, stateful firewall rules, or IDS screens. The service set must be configured identically on each chassis of the high availability pair. The NAT rules, stateful firewall rule, and IDS screens must also be configured identically on each chassis.

For example:

```
user@host#set service-set internal-nat next-hop-service inside-service-interface ams0.100
user@host#set service-set internal-nat next-hop-service outside-service-interface ams0.1000
user@host#set service-set internal-nat next-hop-service nat-rules internal-nat1
```

6. Repeat these steps for the other chassis of the high availability pair.

Inter-Chassis Services Redundancy Overview for Next Gen Services

IN THIS SECTION

- [Introduction to Inter-Chassis Services Redundancy | 485](#)
- [Benefits | 485](#)
- [Services Redundancy Components | 485](#)
- [Services Redundancy Operation | 486](#)

Introduction to Inter-Chassis Services Redundancy

Interchassis redundancy for services is controlled by the services redundancy daemon (SRD). The SRD lets you specify events that trigger a switchover between the primary and standby services PICs, which are on two different MX Series chassis. The SRD monitors conditions, and performs a switchover when an event occurs. Inter-chassis services redundancy is a primary-secondary model, not an active-active cluster. Only one services PIC in a redundancy pair, the current primary, receives traffic to be serviced.

You can configure redundancy based on the following monitored events:

- Link down events.
- FPC and PIC reboots.
- Routing protocol daemon (rpd) terminates and restarts.
- Peer gateway events, including requests to acquire or release primary role, or to broadcast warnings.

Benefits

Inter-chassis services redundancy provides automatic switchovers from a services PIC on one chassis to a services PIC on another chassis when a monitored event occurs.

Services Redundancy Components

The following configurable components control services redundancy processing:

- **Redundancy Event**—A monitored critical event that triggers the redundancy peers to acquire or release primary role or to create a warning, and to add or delete signal routes.

One monitored interface can be part of only one redundancy event, but one redundancy event can have multiple monitored interfaces.

- **Redundancy Policy**—A policy that defines the set of actions taken when a redundancy event occurs. Available actions include acquisition or release of primary role, creation of a warning, and addition or deletion of signal routes. You can configure a maximum of 256 redundancy policies. A redundancy policy can have a maximum of 256 interface-down events.

One redundancy event can be part of only one redundancy policy, but one redundancy policy can have multiple redundancy events. For example, redundancy policy RP1 can include redundancy events RE1 and RE2. Redundancy events RE1 and RE2 cannot be included in redundancy policies other than RP1.

- **Redundancy Set**—A collection of one or more redundancy policies that is assigned to one or more service sets on each MX Series chassis of the redundant pair, and the redundancy group that is associated with the redundancy set. At a given time, a particular redundancy set can be active on only one gateway, but not all redundancy sets have to be active on the same gateway. For example, redundancy set A can be active on gateway 1 while redundancy set B is active on gateway 2. You can configure a maximum of 128 redundancy sets.

One service set can be assigned only one redundancy set, but multiple service sets can be assigned the same redundancy set.

One redundancy policy can be part of only one redundancy set, but one redundancy set can have multiple redundancy policies. For example, redundancy set RS1 can include redundancy policies RP1 and RP2. Redundancy policies RP1 and RP2 cannot be included in redundancy sets other than RS1. A redundancy set can have a maximum of 16 redundancy policies.

- **Redundancy Group**—The redundancy group identifies the associated ICCP redundancy group. A one-to-one relationship exists between a redundancy set and a redundancy group. One redundancy set can be part of only one redundancy group. You can configure a maximum of 16 redundancy groups. A maximum of 16 redundancy sets can be associated with the same redundancy group.
- **Signal routes**—Static routes that are added or deleted by services redundancy processing, based on primary role state changes.
- **Routing Policies**—Policies that advertise routes based on the existence or non-existence of signal routes.
- **VRRP (Virtual Router Redundancy Protocol) route tracking**—Tracks whether a reachable signal route exists in the routing table of the routing instance in the configuration. Based on the reachability of the tracked route, VRRP route tracking dynamically changes the priority of the VRRP group.

Services Redundancy Operation

Services redundancy operates as follows:

1. The services redundancy daemon runs on the Routing Engine. It continuously monitors configured redundancy events.
2. When a redundancy event is detected, the services redundancy daemon:
 - a. Adds or removes signal routes specified in the redundancy policy.
 - b. Switches services to the standby.
 - c. Updates stateful synchronization roles as needed.
3. Resulting route changes cause:
 - a. The routing policy connected to this route to advertise routes differently.
 - b. VRRP to change advertised priorities.

To summarize the switchover process:

1. A critical event occurs.
2. The services redundancy daemon adds or removes a signal route.
3. A routing policy advertises routes differently. VRRP changes advertised priorities.
4. Services switch over to the standby.
5. Stateful synchronization is updated accordingly.



NOTE: The order of routing priorities must match the order of services primary role.

If a redundancy policy action is release-primary role and the redundancy peer's state is wait, the primary-role-release fails. If a redundancy policy action is release-primary role-force, the primary role release succeeds even if the redundancy peer's state is warned.

Similarly, if a redundancy policy action on the standby is acquire-primary role and the local state is wait, the primary-role-release fails. If a redundancy policy action is acquire-primary role-force, the primary role release succeeds even if the standby state is wait.

You can also use a manual command to trigger a redundancy policy that releases or acquires primary role.

If gateway 1, the chassis that is configured with the lower IP address, is the primary chassis and you deactivate the services redundancy daemon on it, a switchover to gateway 2 occurs. If gateway 2, the chassis that is configured with the higher IP address, is the primary chassis and you deactivate the services redundancy daemon on it, a switchover does not occur.

RELATED DOCUMENTATION

[Configuring Inter-Chassis Services Redundancy for Next Gen Services | 488](#)

Configuring Inter-Chassis Services Redundancy for Next Gen Services

IN THIS SECTION

- [Configuring Non-Stop Services Redundancy for Next Gen Services Service Set | 488](#)
- [Configuring One-Way Services Redundancy for Next Gen Services Service Set | 495](#)

This topic describes how to configure interchassis-services redundancy for Next Gen Services. This topic contains a procedure for configuring non-stop services redundancy (automatic switchovers in both directions) and a procedure for one-way redundancy (automatic switchovers only from the original primary to the original standby).

You can also use a manual request command to release or acquire primary role:

```
request services redundancy-set redundancy-set trigger redundancy-event event-name <force>
```

The command automatically triggers the specified redundancy event. You must create a configuration that assigns the redundancy event to a redundancy policy that either releases or acquires primary role. You must also assign the redundancy policy to the redundancy set used in the command.

Configuring Non-Stop Services Redundancy for Next Gen Services Service Set

Non-stop services redundancy gives you automatic services switchovers between the MX Series routers when a critical event occurs. Automatic switchovers from gateway1 to gateway2 and from gateway2 to gateway1 take place without manual intervention.

To configure non-stop services redundancy for a service set, perform the following steps on both gateway1 and gateway2:

1. Configure one or more redundancy events to monitor the conditions that trigger a services switchover to the peer gateway.

- a. Configure a name for the redundancy event.

```
[edit services]
user@host# set event-options redundancy-event event-name
```

For example:

```
[edit services]
user@host# set event-options redundancy-event RELS_MSHIP_CRIT_EV
```

- b. Specify any interfaces that trigger a services switchover when the interface goes down.

```
[edit services event-options redundancy-event event-name]
user@host# set monitor link-down [interface-name]
```

- c. Specify that a process routing daemon restart request triggers a services switchover.

```
[edit services event-options redundancy-event event-name]
user@host# set monitor process routing restart
```

- d. Specify that a process routing daemon terminate request triggers a services switchover.

```
[edit services event-options redundancy-event event-name]
user@host# set monitor process routing abort
```

- e. Specify that a request from the peer to acquire ownership triggers a services switchover.

```
[edit services event-options redundancy-event event-name]
user@host# set monitor peer mastership-acquire
```

2. Configure a redundancy policy that releases primary role and deletes a static route when the redundancy event conditions are met.

- a. Configure a name for the policy.

```
user@host# edit policy-options redundancy-policy policy-name
```

For example:

```
user@host# edit policy-options redundancy-policy RLS_MSHIP_POL
```

- b. Specify the redundancy events that release primary role.

```
[edit policy-options redundancy-policy policy-name]  
user@host# set redundancy-events [event-list]
```

For example:

```
[edit policy-options redundancy-policy RLS_MSHIP_POL  
user@host# set redundancy-events RELS_MSHIP_CRIT_EV
```

If you want to be able to run the request services `redundancy-set` *redundancy-set* trigger redundancy-event *event-name* <force> to manually release primary role, include that *event-name* in the redundancy policy. The redundancy event itself does not need to be configured, because it is triggered by the request command.

For example:

```
[edit policy-options redundancy-policy RLS_MSHIP_POL  
user@host# set redundancy-events [RELS_MSHIP_CRIT_EV RELS_MSHIP_MANUAL_EV]
```

- c. Release primary role.

```
[edit policy-options redundancy-policy policy-name]  
user@host# set then release-mastership
```

- d. Delete the static route.

```
[edit policy-options redundancy-policy policy-name]  
user@host# set then delete-static-route destination (receive | next-hop next-hop) routing-  
instance routing-instance
```

3. Configure a redundancy event to identify when the peer gateway releases primary role.

```
[edit services]
user@host# set event-options redundancy-event event-name monitor peer release-mastership
```

For example:

```
[edit services]
user@host# set event-options redundancy-event PEER_RELS_MSHIP_EV monitor peer release-
mastership
```

4. Configure a redundancy policy that acquires primary role from the peer gateway and adds a static route.
 - a. Configure a name for the policy.

```
user@host# edit policy-options redundancy-policy policy-name
```

For example:

```
user@host# edit policy-options redundancy-policy ACQU_MSHIP_POL
```

- b. Specify the redundancy events that acquire primary role.

```
[edit policy-options redundancy-policy policy-name]
user@host# set redundancy-events [event-list]
```

For example:

```
[edit policy-options redundancy-policy ACQU_MSHIP_POL]
user@host# set redundancy-events PEER_RELS_MSHIP_EV
```

If you want to be able to run the request services `redundancy-set redundancy-set trigger redundancy-event event-name <force>` to manually acquire primary role, include that *event-name* in the redundancy policy. The redundancy event itself does not need to be configured, because it is triggered by the request command.

For example:

```
[edit policy-options redundancy-policy ACQU_MSHIP_POL]
user@host# set redundancy-events [PEER_RELS_MSHIP_EV ACQU_MSHIP_MANUAL_EV]
```

- c. Acquire primary role.

```
[edit policy-options redundancy-policy policy-name]
user@host# set then acquire-mastership
```

- d. Add a static route.

```
[edit policy-options redundancy-policy policy-name]
user@host# set then add-static-route destination (receive | next-hop next-hop) routing-
instance routing-instance
```

5. Configure the redundancy set.

- a. Configure a name for the redundancy set.

```
[edit services]
user@host# set redundancy-set redundancy-set
```

For example:

```
[edit services]
user@host# set redundancy-set 1
```

- b. Specify the redundancy group ID for the redundancy set.

```
[edit services redundancy-set redundancy-set]
user@host# set redundancy-group redundancy-group
```

For example:

```
[edit services redundancy-set 1]
user@host# set redundancy-group 1
```

The redundancy group ID is the same redundancy group ID configured for the ICCP daemon (iccpd) through the existing ICCP configuration hierarchy. For example,

```
iccp {
    local-ip-addr 10.1.1.1;
    peer 10.2.2.2 {
        redundancy-group-id-list 1;
        liveness-detection {
            minimum-interval 1000;
        }
    }
}
```

- c. Specify the redundancy policy that releases primary role and the redundancy policy that acquires primary role.

```
[edit services redundancy-set redundancy-set]
user@host# set redundancy-policy [redundancy-policy-list]
```

For example:

```
[edit services redundancy-set 1]
user@host# set redundancy-policy [ACQU_MSHIP_POL RLS_MSHIP_POL]
```

- d. Configure the frequency of health check probes of the redundancy set, in seconds.

```
[edit services redundancy-set redundancy-set]
user@host# set healthcheck-timer-interval healthcheck-timer-interval
```

The default is 30 seconds.

- e. Configure the maximum wait time for a help check response, in seconds.

```
[edit services redundancy-set redundancy-set]
user@host# set hold-time hold-time
```

The range is 0 through 3600 seconds.

- f. Configure the frequency of srd hello messages, in seconds.

```
[edit services redundancy-set redundancy-set]
user@host# set keepalive keepalive
```

The range is 1 through 60 seconds.

6. Configure routing policies.

- a. Identify signal routes that requires redundancy-related routing changes. Specify the signal route and the routing table that is used.

```
[edit policy-options condition condition-name]
user@host# set if-route-exists signal-route table routing-table
```

For example:

```
[edit policy-options condition switchover-route-exists]
user@host# set if-route-exists 10.45.45.0/24 table bgp1_table
```

- b. To change the local-preference for the signal route, enter it in a policy statement.

```
[edit policy-options policy-statement policy-name]
user@host# set term term from protocol [protocol variables] prefix-list prefix-list
condition condition-name then local-preference preference-value accept
```

- c. To change as-path-prepend values for the signal route, enter them in the policy statement.

```
[edit policy-options policy-statement policy-name]
user@host# set term term from prefix-list prefix-list condition condition-name then as-
path-prepend [as-prepend-values] next-hop self accept
```

7. Configure redundancy for the service set by assigning the redundancy set to the service set.

```
[edit]
user@host# set services service-set service-set-name redundancy-set-id redundancy-set
```

8. Repeat these steps on the peer gateway.

SEE ALSO

[Configuring One-Way Services Redundancy for Next Gen Services Service Set](#)

Configuring One-Way Services Redundancy for Next Gen Services Service Set

One-way services redundancy gives you automatic services switchovers from gateway1, the original primary gateway, to gateway2, the original standby gateway. An automatic switchover from gateway 2 to gateway1 does not happen. To switchover from gateway2 to gateway1, you must perform a manual switchover.

1. On gateway1, the initial primary, configure one or more redundancy events to monitor the conditions that trigger a services switchover to gateway2, the standby gateway.
 - a. Configure a name for the redundancy event.

```
[edit services]
user@gateway1# set event-options redundancy-event event-name
```

For example:

```
[edit services]
user@gateway1# set event-options redundancy-event RELS_MSHIP_CRIT_EV
```

- b. Specify any interfaces that trigger a services switchover when the interface goes down.

```
[edit services event-options redundancy-event event-name]
user@gateway1# set monitor link-down [interface-name]
```

- c. Specify that a process routing daemon restart request triggers a services switchover.

```
[edit services event-options redundancy-event event-name]
user@gateway1# set monitor process routing restart
```

- d. Specify that a process routing daemon terminate request triggers a services switchover.

```
[edit services event-options redundancy-event event-name]
user@gateway1# set monitor process routing abort
```

2. On gateway1, configure a redundancy policy that releases primary role and deletes a static route when the redundancy event conditions are met.
 - a. Configure a name for the policy.

```
user@gateway1# edit policy-options redundancy-policy policy-name
```

For example:

```
user@gateway1# edit policy-options redundancy-policy RLS_MSHIP_POL
```

- b. Specify the redundancy events that release primary role.

```
[edit policy-options redundancy-policy policy-name]
user@gateway1# set redundancy-events [event-list]
```

For example:

```
[edit policy-options redundancy-policy RLS_MSHIP_POL]
user@gateway1# set redundancy-events RELS_MSHIP_CRIT_EV
```

If you want to be able to run the request services `redundancy-set redundancy-set` trigger redundancy-event `event-name <force>` to manually release primary role, include that `event-name` in the redundancy policy. The redundancy event itself does not need to be configured, because it is triggered by the request command.

For example:

```
[edit policy-options redundancy-policy RLS_MSHIP_POL]
user@gateway1# set redundancy-events [RELS_MSHIP_CRIT_EV RELS_MSHIP_MANUAL_EV]
```

- c. Release primary role.

```
[edit policy-options redundancy-policy policy-name]
user@gateway1# set then release-mastership force
```

- d. Delete the static route.

```
[edit policy-options redundancy-policy policy-name]
user@gateway1# set then delete-static-route destination (receive | next-hop next-hop)
routing-instance routing-instance
```

3. On gateway1, configure a redundancy policy that acquires primary role from gateway2 when you perform a manual request on gateway1 (request services redundancy-set *redundancy-set* trigger redundancy-event *event-name* <force>).
 - a. Configure a name for the policy.

```
user@gateway1# edit policy-options redundancy-policy policy-name
```

For example:

```
user@gateway1# edit policy-options redundancy-policy ACQU_MSHIP_POL
```

- b. Specify the name of the redundancy event that the manual request uses.

```
[edit policy-options redundancy-policy policy-name]
user@gateway1# set redundancy-events event-name
```

For example:

```
[edit policy-options redundancy-policy ACQU_MSHIP_POL]
user@gateway1# set redundancy-events ACQU_MSHIP_MANUAL_EV
```

The redundancy event itself does not need to be configured, because it is triggered by the request command.

- c. Acquire primary role.

```
[edit policy-options redundancy-policy policy-name]
user@host# set then acquire-mastership
```

4. On gateway1, configure the redundancy set.

- a. Configure a name for the redundancy set.

```
[edit services]
user@gateway1# set redundancy-set redundancy-set
```

For example:

```
[edit services]
user@gateway1# set redundancy-set 1
```

- b. Specify the redundancy group ID for the redundancy set.

```
[edit services redundancy-set redundancy-set]
user@gateway1# set redundancy-group redundancy-group
```

For example:

```
[edit services redundancy-set 1]
user@gateway1# set redundancy-group 1
```

The redundancy group ID is the same redundancy group ID configured for the ICCP daemon (iccpd) through the existing ICCP configuration hierarchy. For example,

```
iccp {
  local-ip-addr 10.1.1.1;
  peer 10.2.2.2 {
    redundancy-group-id-list 1;
    liveness-detection {
      minimum-interval 1000;
    }
  }
}
```

- c. Specify the redundancy policy that releases primary role and the redundancy policy that acquires primary role.

```
[edit services redundancy-set redundancy-set]
user@gateway1# set redundancy-policy [redundancy-policy-list]
```

For example:

```
[edit services redundancy-set 1]
user@gateway1# set redundancy-policy [ ACQU_MSHIP_POL RLS_MSHIP_POL]
```

- d. Configure the frequency of health check probes of the redundancy set, in seconds.

```
[edit services redundancy-set redundancy-set]
user@gateway1# set healthcheck-timer-interval healthcheck-timer-interval
```

The default is 30 seconds.

- e. Configure the maximum wait time for a health check response, in seconds.

```
[edit services redundancy-set redundancy-set]
user@gateway1# set hold-time hold-time
```

The range is 0 through 3600 seconds.

- f. Configure the frequency of srd hello messages, in seconds.

```
[edit services redundancy-set redundancy-set]
user@gateway1# set keepalive keepalive
```

The range is 1 through 60 seconds.

5. On gateway1, configure routing policies.

- a. Identify signal routes that requires redundancy-related routing changes. Specify the signal route and the routing table that is used.

```
[edit policy-options condition condition-name]
user@gateway1# set if-route-exists signal-route table routing-table
```


For example:

```
[edit policy-options condition switchover-route-exists]
user@gateway1# set if-route-exists 10.45.45.0/24 table bgp1_table
```

- b. To change the local-preference for the signal route, enter it in a policy statement.

```
[edit policy-options policy-statement policy-name]
user@gateway1# set term term from protocol [protocol variables] prefix-list prefix-list
condition condition-name then local-preference preference-value accept
```

- c. To change as-path-prepend values for the signal route, enter them in the policy statement.

```
[edit policy-options policy-statement policy-name]
user@gateway1# set term term from prefix-list prefix-list condition condition-name then
as-path-prepend [as-prepend-values] next-hop self accept
```

6. On gateway1, configure redundancy for the service set by assigning the redundancy set to the service set.

```
[edit]
user@gateway1# set services service-set service-set-name redundancy-set-id redundancy-set
```

7. On gateway2, the initial standby, configure a redundancy event to identify when the peer gateway releases primary role.

```
[edit services]
user@gateway2# set event-options redundancy-event event-name monitor peer release-mastership
```

For example:

```
[edit services]
user@gateway2# set event-options redundancy-event PEER_RELS_MSHIP_EV monitor peer release-
mastership
```

8. On gateway2, configure a redundancy policy that acquires primary role from the peer gateway and adds a static route.

- a. Configure a name for the policy.

```
user@gateway2# edit policy-options redundancy-policy policy-name
```

For example:

```
user@gateway2# edit policy-options redundancy-policy ACQU_MSHIP_POL
```

- b. Specify the configured redundancy event for the peer gateway primary role release event.

```
[edit policy-options redundancy-policy policy-name]  
user@gateway2# set redundancy-events event-name
```

For example:

```
[edit policy-options redundancy-policy ACQU_MSHIP_POL]  
user@gateway2# set redundancy-events PEER_RELS_MSHIP_EV
```

- c. Acquire primary role.

```
[edit policy-options redundancy-policy policy-name]  
user@gateway2# set then acquire-mastership
```

- d. Add a static route.

```
[edit policy-options redundancy-policy policy-name]  
user@gateway2# set then add-static-route destination (receive | next-hop next-hop)  
routing-instance routing-instance
```

9. On gateway2, configure a redundancy event to identify when the peer gateway requests primary role.

```
[edit services]  
user@gateway2# set event-options redundancy-event event-name monitor peer mastership-acquire
```

For example:

```
[edit services]
user@gateway2# set event-options redundancy-event PEER_MSHIP_ACQU_EV monitor peer
mastership-acquire
```

10. On gateway2, configure a redundancy policy that releases primary role and deletes a static route when gateway1 requests primary role.
 - a. Configure a name for the policy.

```
user@gateway2# edit policy-options redundancy-policy policy-name
```

For example:

```
user@gateway2# edit policy-options redundancy-policy RELS-MSHIP_POL
```

- b. Specify the configured redundancy event that identifies when the peer gateway requests primary role.

```
[edit policy-options redundancy-policy policy-name]
user@gateway2# set redundancy-events event-name
```

For example:

```
[edit policy-options redundancy-policy RELS-MSHIP_POL]
user@gateway2# set redundancy-events PEER_MSHIP_ACQU_EV
```

- c. Release primary role.

```
[edit policy-options redundancy-policy policy-name]
user@gateway2# set then release-mastership force
```

- d. Delete the static route.

```
[edit policy-options redundancy-policy policy-name]
user@gateway2# set then delete-static-route destination (receive | next-hop next-hop)
routing-instance routing-instance
```

11. On gateway2, configure one or more redundancy events to monitor the conditions that trigger a warning.

- a. Configure a name for the redundancy event.

```
[edit services]
user@gateway2# set event-options redundancy-event event-name
```

For example:

```
[edit services]
user@gateway2# set event-options redundancy-event WARN_EV
```

- b. Specify any interfaces that trigger a warning when the interface goes down.

```
[edit services event-options redundancy-event event-name]
user@gateway2# set monitor link-down [interface-name]
```

- c. Specify that a process routing daemon restart request triggers a warning.

```
[edit services event-options redundancy-event event-name]
user@gateway2# set monitor process routing restart
```

- d. Specify that a process routing daemon terminate request triggers a warning.

```
[edit services event-options redundancy-event event-name]
user@gateway2# set monitor process routing abort
```

12. On gateway2, configure a redundancy policy that broadcasts a warning.

- a. Configure a name for the policy.

```
user@gateway2# edit policy-options redundancy-policy policy-name
```

For example:

```
user@gateway2# edit policy-options redundancy-policy WARN_POL
```

- b. Specify the configured redundancy events that trigger a warning.

```
[edit policy-options redundancy-policy policy-name]  
user@gateway2# set redundancy-events [event-list]
```

For example:

```
[edit policy-options redundancy-policy WARN_POL]  
user@gateway2# set redundancy-events WARN_EV
```

- c. Broadcast the warning.

```
[edit policy-options redundancy-policy policy-name]  
user@gateway2# set then broadcast-warning
```

13. On gateway2, configure the redundancy set.

- a. Configure a name for the redundancy set.

```
[edit services]  
user@gateway2# set redundancy-set redundancy-set
```

For example:

```
[edit services]  
user@gateway2# set redundancy-set 1
```

- b. Specify the redundancy group ID for the redundancy set.

```
[edit services redundancy-set redundancy-set]
user@gateway2# set redundancy-group redundancy-group
```

For example:

```
[edit services redundancy-set 1]
user@gateway2# set redundancy-group 1
```

The redundancy group ID is the same redundancy group ID configured for the ICCP daemon (iccpd) through the existing ICCP configuration hierarchy. For example,

```
iccp {
  local-ip-addr 10.1.1.1;
  peer 10.2.2.2 {
    redundancy-group-id-list 1;
    liveness-detection {
      minimum-interval 1000;
    }
  }
}
```

- c. Specify the redundancy policy that releases primary role, the redundancy policy that acquires primary role, and the redundancy policy that triggers a warning.

```
[edit services redundancy-set redundancy-set]
user@gateway2# set redundancy-policy [redundancy-policy-list]
```

For example:

```
[edit services redundancy-set 1]
user@gateway2# set redundancy-policy [ ACQU_MSHIP_POL RLS_MSHIP_POL WARN_POL]
```

- d. Configure the frequency of health check probes of the redundancy set, in seconds.

```
[edit services redundancy-set redundancy-set]
user@gateway2# set healthcheck-timer-interval healthcheck-timer-interval
```

The default is 30 seconds.

- e. Configure the maximum wait time for a health check response, in seconds.

```
[edit services redundancy-set redundancy-set]
user@gateway2# set hold-time hold-time
```

The range is 0 through 3600 seconds.

- f. Configure the frequency of srp hello messages, in seconds.

```
[edit services redundancy-set redundancy-set]
user@gateway2# set keepalive keepalive
```

The range is 1 through 60 seconds.

14. On gateway2, configure routing policies.

- a. Identify signal routes that requires redundancy-related routing changes. Specify the signal route and the routing table that is used.

```
[edit policy-options condition condition-name]
user@gateway2# set if-route-exists signal-route table routing-table
```

For example:

```
[edit policy-options condition switchover-route-exists]
user@gateway2# set if-route-exists 10.45.45.0/24 table bgp1_table
```

- b. To change the local-preference for the signal route, enter it in a policy statement.

```
[edit policy-options policy-statement policy-name]
user@gateway2# set term term from protocol [protocol variables] prefix-list prefix-list
condition condition-name then local-preference preference-value accept
```

- c. To change as-path-prepend values for the signal route, enter them in the policy statement.

```
[edit policy-options policy-statement policy-name]  
user@gateway2# set term term from prefix-list prefix-list condition condition-name then  
as-path-prepend [as-prepend-values] next-hop self accept
```

15. On gateway2, configure redundancy for the service set by assigning the redundancy set to the service set.

```
[edit]  
user@gateway2# set services service-set service-set-name redundancy-set-id redundancy-set
```

SEE ALSO

| [Inter-Chassis Services Redundancy Overview for Next Gen Services](#) | 485

11

PART

Application Layer Gateways

- [Enabling Traffic to Pass Securely Using Application Layer Gateways | 509](#)
-

Enabling Traffic to Pass Securely Using Application Layer Gateways

IN THIS CHAPTER

- [Next Gen Services Application Layer Gateways | 509](#)
- [Configuring Application Sets | 519](#)
- [Configuring Application Properties for Next Gen Services | 520](#)
- [Examples: Configuring Application Protocols | 537](#)
- [Verifying the Output of ALG Sessions | 538](#)

Next Gen Services Application Layer Gateways

IN THIS SECTION

- [RTSP | 509](#)
- [SIP | 510](#)
- [Configuring SIP | 510](#)

This topic describes the Application Layer Gateways (ALGs) supported by Junos OS for Next Gen Services. ALG support includes managing pinholes and parent-child relationships for the supported ALGs.

RTSP

The Real-Time Streaming Protocol (RTSP) controls the delivery of data with real-time properties such as audio and video. The streams controlled by RTSP can use RTP, but it is not required. Media can be transmitted on the same RTSP control stream. This is an HTTP-like text-based protocol, but client and

server maintain session information. A session is established using the SETUP message and terminated using the TEARDOWN message. The transport (the media protocol, address, and port numbers) is negotiated in the setup and the setup-response.

Support for stateful firewall and NAT services requires that you configure the RTSP ALG for TCP port 554.

The ALG monitors the control connection, opens flows dynamically for media (RTP/RTSP) streams, and performs NAT address and port rewrites.

SIP

The Session Initiation Protocol (SIP) is an application layer protocol that can establish, maintain, and terminate media sessions. It is a widely used voice over IP (VoIP) signaling protocol. The SIP ALG monitors SIP traffic and dynamically creates and manages pinholes on the signaling and media paths. The ALG only allows packets with the correct permissions. The SIP ALG also performs the following functions:

- Manages parent-child session relationships.
- Enforces security policies.
- Manages pinholes for VoIP traffic.

The SIP ALG supports the following features:

- Stateful firewall
- Static source NAT
- Dynamic address only source NAT
- *Network Address Port Translation (NAPT)*



NOTE: SIP sessions are limited to 12 hours (720 minutes) for NAT processing on the MS-MIC and MS-MPC interface cards. SIP sessions on the MS-DPC have no time limit.

Configuring SIP

The Session Initiation Protocol (SIP) is a generalized protocol for communication between endpoints involved in Internet services such as telephony, fax, video conferencing, instant messaging, and file exchange.

The Junos OS provides ALG services in accordance with the standard described in RFC 3261, *SIP: Session Initiation Protocol*. SIP flows under the Junos OS are as described in RFC 3665, *Session Initiation Protocol (SIP) Basic Call Flow Examples*.



NOTE: Before implementing the Junos OS SIP ALG, you should be familiar with certain limitations, discussed in ["Junos OS SIP ALG Limitations" on page 518](#)

The use of NAT in conjunction with the SIP ALG results in changes in SIP header fields due to address translation. For an explanation of these translations, refer to ["SIP ALG Interaction with Network Address Translation" on page 512](#).

To implement SIP on adaptive services interfaces, you configure the `application-protocol` statement at the `[edit applications application application-name]` hierarchy level with the value `sip`. In addition, there are two other statements you can configure to modify how SIP is implemented:

- You can enable the router to accept any incoming SIP calls for the endpoint devices that are behind the NAT firewall. When a device behind the firewall registers with the proxy that is outside the firewall, the AS or Multiservices PIC maintains the registration state. When the `learn-sip-register` statement is enabled, the router can use this information to accept inbound calls. If this statement is not configured, no inbound calls are accepted; only the devices behind the firewall can call devices outside the firewall.

To configure SIP registration, include the `learn-sip-register` statement at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]  
learn-sip-register;
```



NOTE: The `learn-sip-register` statement is not applicable to the Next Gen Services MX-SPC3.

You can also manually inspect the SIP register by issuing the `show services stateful-firewall sip-register` command; for more information, see the *Junos OS System Basics and Services Command Reference*. The `show services stateful-firewall sip-register` command is not supported for Next Gen Services.

- You can specify a timeout period for the duration of SIP calls that are placed on hold. When a call is put on hold, there is no activity and flows might time out after the configured `inactivity-timeout` period expires, resulting in call state teardown. To avoid this, when a call is put on hold, the flow timer is reset to the `sip-call-hold-timeout` cycle to preserve the call state and flows for longer than the `inactivity-timeout` period.



NOTE: The sip-call-hold-timeout statement is not applicable to the Next Gen Services MX-SPC3.

To configure a timeout period, include the sip-call-hold-timeout statement at the [edit applications application *application-name*] hierarchy level:

```
[edit applications application application-name]
sip-call-hold-timeout seconds;
```

The default value is 7200 seconds and the range is from 0 through 36,000 seconds (10 hours).

SIP ALG Interaction with Network Address Translation

The Network Address Translation (NAT) protocol enables multiple hosts in a private subnet to share a single public IP address to access the Internet. For outgoing traffic, NAT replaces the private IP address of the host in the private subnet with the public IP address. For incoming traffic, the public IP address is converted back into the private address, and the message is routed to the appropriate host in the private subnet.

Using NAT with the Session Initiation Protocol (SIP) service is more complicated because SIP messages contain IP addresses in the SIP headers as well as in the SIP body. When using NAT with the SIP service, the SIP headers contain information about the caller and the receiver, and the device translates this information to hide it from the outside network. The SIP body contains the Session Description Protocol (SDP) information, which includes IP addresses and port numbers for transmission of the media. The device translates SDP information for allocating resources to send and receive the media.

How IP addresses and port numbers in SIP messages are replaced depends on the direction of the message. For an outgoing message, the private IP address and port number of the client are replaced with the public IP address and port number of the Juniper Networks firewall. For an incoming message, the public address of the firewall is replaced with the private address of the client.

When an INVITE message is sent out across the firewall, the SIP Application Layer Gateway (ALG) collects information from the message header into a call table, which it uses to forward subsequent messages to the correct endpoint. When a new message arrives, for example an ACK or 200 OK, the ALG compares the "From:, To:, and Call-ID:" fields against the call table to identify the call context of the message. If a new INVITE message arrives that matches the existing call, the ALG processes it as a REINVITE.

When a message containing SDP information arrives, the ALG allocates ports and creates a NAT mapping between them and the ports in the SDP. Because the SDP requires sequential ports for the Real-Time Transport Protocol (RTP) and Real-Time Control Protocol (RTCP) channels, the ALG provides consecutive even-odd ports. If it is unable to find a pair of ports, it discards the SIP message.

This topic contains the following sections:

Outgoing Calls

When a SIP call is initiated with a SIP request message from the internal to the external network, NAT replaces the IP addresses and port numbers in the SDP and binds the IP addresses and port numbers to the Juniper Networks firewall. Via, Contact, Route, and Record-Route SIP header fields, if present, are also bound to the firewall IP address. The ALG stores these mappings for use in retransmissions and for SIP response messages.

The SIP ALG then opens pinholes in the firewall to allow media through the device on the dynamically assigned ports negotiated based on information in the SDP and the Via, Contact, and Record-Route header fields. The pinholes also allow incoming packets to reach the Contact, Via, and Record-Route IP addresses and ports. When processing return traffic, the ALG inserts the original Contact, Via, Route, and Record-Route SIP fields back into packets.

Incoming Calls

Incoming calls are initiated from the public network to public static NAT addresses or to interface IP addresses on the device. Static NATs are statically configured IP addresses that point to internal hosts; interface IP addresses are dynamically recorded by the ALG as it monitors REGISTER messages sent by internal hosts to the SIP registrar. When the device receives an incoming SIP packet, it sets up a session and forwards the payload of the packet to the SIP ALG.

The ALG examines the SIP request message (initially an INVITE) and, based on information in the SDP, opens gates for outgoing media. When a 200 OK response message arrives, the SIP ALG performs NAT on the IP addresses and ports and opens pinholes in the outbound direction. (The opened gates have a short time-to-live, and they time out if a 200 OK response message is not received quickly.)

When a 200 OK response arrives, the SIP proxy examines the SDP information and reads the IP addresses and port numbers for each media session. The SIP ALG on the device performs NAT on the addresses and port numbers, opens pinholes for outbound traffic, and refreshes the timeout for gates in the inbound direction.

When the ACK arrives for the 200 OK, it also passes through the SIP ALG. If the message contains SDP information, the SIP ALG ensures that the IP addresses and port numbers are not changed from the previous INVITE—if they are, the ALG deletes old pinholes and creates new pinholes to allow media to pass through. The ALG also monitors the Via, Contact, and Record-Route SIP fields and opens new pinholes if it determines that these fields have changed.

Forwarded Calls

A forwarded call is when, for example, user A outside the network calls user B inside the network, and user B forwards the call to user C outside the network. The SIP ALG processes the INVITE from user A

as a normal incoming call. But when the ALG examines the forwarded call from B to C outside the network and notices that B and C are reached using the same interface, it does not open pinholes in the firewall, because media will flow directly between user A and user C.

Call Termination

The BYE message terminates a call. When the device receives a BYE message, it translates the header fields just as it does for any other message. But because a BYE message must be acknowledged by the receiver with a 200 OK, the ALG delays call teardown for five seconds to allow time for transmission of the 200 OK.

Call Re-INVITE Messages

Re-INVITE messages add new media sessions to a call and remove existing media sessions. When new media sessions are added to a call, new pinholes are opened in the firewall and new address bindings are created. The process is identical to the original call setup. When one or more media sessions are removed from a call, pinholes are closed and bindings released just as with a BYE message.

Call Session Timers

The SIP ALG uses the Session-Expires value to time out a session if a Re-INVITE or UPDATE message is not received. The ALG gets the Session-Expires value, if present, from the 200 OK response to the INVITE and uses this value for signaling timeout. If the ALG receives another INVITE before the session times out, it resets all timeout values to this new INVITE or to default values, and the process is repeated.

As a precautionary measure, the SIP ALG uses hard timeout values to set the maximum amount of time a call can exist. This ensures that the device is protected should one of the following events occur:

- End systems crash during a call and a BYE message is not received.
- Malicious users never send a BYE in an attempt to attack a SIP ALG.
- Poor implementations of SIP proxy fail to process Record-Route and never send a BYE message.
- Network failures prevent a BYE message from being received.

Call Cancellation

Either party can cancel a call by sending a CANCEL message. Upon receiving a CANCEL message, the SIP ALG closes pinholes through the firewall—if any have been opened—and releases address bindings. Before releasing the resources, the ALG delays the control channel age-out for approximately five seconds to allow time for the final 200 OK to pass through. The call is terminated when the five second timeout expires, regardless of whether a 487 or non-200 response arrives.

Forking

Forking enables a SIP proxy to send a single INVITE message to multiple destinations simultaneously. When the multiple 200 OK response messages arrive for the single call, the SIP ALG parses but updates call information with the first 200 OK messages it receives.

SIP Messages

The SIP message format consists of a SIP header section and the SIP body. In request messages, the first line of the header section is the request line, which includes the method type, request-URI, and protocol version. In response messages, the first line is the status line, which contains a status code. SIP headers contain IP addresses and port numbers used for signaling. The SIP body, separated from the header section by a blank line, is reserved for session description information, which is optional. Junos OS currently supports the SDP only. The SIP body contains IP addresses and port numbers used to transport the media.

SIP Headers

In the following sample SIP request message, NAT replaces the IP addresses in the header fields to hide them from the outside network.

```
INVITE bob@10.150.20.5 SIP/2.0
Via: SIP/2.0/UDP 10.150.20.3:5434
From: alice@10.150.20.3
To: bob@10.150.20.5
Call-ID: a12abcde@10.150.20.3
Contact: alice@10.150.20.3:5434
Route: <sip:netscreen@10.150.20.3:5060>
Record-Route: <sip:netscreen@10.150.20.3:5060>
```

How IP address translation is performed depends on the type and direction of the message. A message can be any of the following:

- Inbound request
- Outbound response
- Outbound request
- Inbound response

[Table 48 on page 516](#) shows how NAT is performed in each of these cases. Note that for several of the header fields the ALG determine more than just whether the messages comes from inside or outside the

network. It must also determine what client initiated the call, and whether the message is a request or response.

Table 48: Requesting Messages with NAT Table

Inbound Request (from public to private)	To:	Replace domain with local address
	From:	None
	Call-ID:	None
	Via:	None
	Request-URI:	Replace ALG address with local address
	Contact:	None
	Record-Route:	None
	Route:	None
Outbound Response (from private to public)	To:	Replace ALG address with local address
	From:	None
	Call-ID:	None
	Via:	None
	Request-URI:	N/A
	Contact:	Replace local address with ALG address
	Record-Route:	Replace local address with ALG address

	Route:	None
Outbound Request (from private to public)	To:	None
	From:	Replace local address with ALG address
	Call-ID:	None
	Via:	Replace local address with ALG address
	Request-URI:	None
	Contact:	Replace local address with ALG address
	Record-Route:	Replace local address with ALG address
	Route:	Replace ALG address with local address
Outbound Response (from public to private)	To:	None
	From:	Replace ALG address with local address
	Call-ID:	None
	Via:	Replace ALG address with local address
	Request-URI:	N/A
	Contact:	None
	Record-Route:	Replace ALG address with local address
	Route:	Replace ALG address with local address

SIP Body

The SDP information in the SIP body includes IP addresses the ALG uses to create channels for the media stream. Translation of the SDP section also allocates resources, that is, port numbers to send and receive the media.

The following excerpt from a sample SDP section shows the fields that are translated for resource allocation.

```
o=user 2344234 55234434 IN IP4 10.150.20.3
c=IN IP4 10.150.20.3
m=audio 43249 RTP/AVP 0
```

SIP messages can contain more than one media stream. The concept is similar to attaching multiple files to an e-mail message. For example, an INVITE message sent from a SIP client to a SIP server might have the following fields:

```
c=IN IP4 10.123.33.4
m=audio 33445 RTP/AVP 0
c=IN IP4 10.123.33.4
m=audio 33447 RTP/AVP 0
c=IN IP4 10.123.33.4
m=audio 33449 RTP/AVP 0
```

Junos OS supports up to 6 SDP channels negotiated for each direction, for a total of 12 channels per call.

Junos OS SIP ALG Limitations

The following limitations apply to configuration of the SIP ALG:

- Only the methods described in RFC 3261 are supported.
- Only SIP version 2 is supported.
- TCP is not supported as a transport mechanism for signaling messages for MS-MPCs but is supported for Next Gen Services.
- *Do not configure the SIP ALG when using STUN.* if clients use STUN/TURN to detect the firewall or NAT devices between the caller and responder or proxy, the client attempts to best-guess the NAT device behavior and act accordingly to place the call.

- On MS-MPCs, do not use the endpoint-independent mapping NAT pool option in conjunction with the SIP ALG. Errors will result. This does not apply to Next Gen Services.
- IPv6 signaling data is not supported for MS-MPCs but is supported for Next Gen Services.
- Authentication is not supported.
- Encrypted messages are not supported.
- SIP fragmentation is not supported for MS-MPCs but is supported for Next Gen Services.
- The maximum UDP packet size containing a SIP message is assumed to be 9 KB. SIP messages larger than this are not supported.
- The maximum number of media channels in a SIP message is assumed to be six.
- Fully qualified domain names (FQDNs) are not supported in critical fields.
- QoS is not supported. SIP supports DSCP rewrites.
- High availability is not supported, except for warm standby.
- A timeout setting of never is not supported on SIP or NAT.
- Multicast (forking proxy) is not supported.

RELATED DOCUMENTATION

ALG Descriptions

ALGs Available for Junos OS Address Aware NAT

Configuring Application Sets

You can group the applications you have defined into a named object by including the `application-set` statement at the `[edit applications]` hierarchy level with an `application` statement for each application:

```
[edit applications]
  application-set application-set-name {
    application application;
  }
```

For an example of a typical application set, see *Examples: Configuring Application Protocols*.

Configuring Application Properties for Next Gen Services

IN THIS SECTION

- [Configuring an Application Protocol | 521](#)
- [Configuring the Network Protocol | 523](#)
- [Configuring the ICMP Code and Type | 525](#)
- [Configuring Source and Destination Ports | 526](#)
- [Configuring the Inactivity Timeout Period | 527](#)
- [Configuring SIP | 527](#)
- [Configuring an SNMP Command for Packet Matching | 536](#)

To configure application properties, include the application statement at the [edit applications] hierarchy level:

```
[edit applications]
application application-name {
    application-protocol protocol-name;
    child-inactivity-timeout seconds;
    destination-port port-number;
    gate-timeout seconds;
    icmp-code value;
    icmp-type value;
    inactivity-timeout value;
    protocol type;
    rpc-program-number number;
    snmp-command command;
    source-port port-number;
    ttl-threshold value;
    uuid hex-value;
}
```

You can group application objects by configuring the application-set statement; for more information, see *Configuring Application Sets*.

This section includes the following tasks for configuring applications:

Configuring an Application Protocol

The `application-protocol` statement allows you to specify which of the supported application protocols (ALGs) to configure and include in an application set for service processing. To configure application protocols, include the `application-protocol` statement at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]
application-protocol protocol-name;
```

[Table 49 on page 521](#) shows the list of supported protocols for Next Gen Services. For more information about specific protocols, see *ALG Descriptions*.

Table 49: Application Protocols Supported by Services Interfaces

Protocol Name	CLI Value	Comments
Bootstrap protocol (BOOTP)	bootp	Supports BOOTP and dynamic host configuration protocol (DHCP).
Distributed Computing Environment (DCE) remote procedure call (RPC)	dce-rpc	Requires the protocol statement to have the value <code>udp</code> or <code>tcp</code> . Requires a <code>uuid</code> value. You cannot specify <code>destination-port</code> or <code>source-port</code> values.
DCE RPC portmap	dce-rpc-portmap	Requires the protocol statement to have the value <code>udp</code> or <code>tcp</code> . Requires a <code>destination-port</code> value.
Domain Name System (DNS)	dns	Requires the protocol statement to have the value <code>udp</code> . This application protocol closes the DNS flow as soon as the DNS response is received.
Exec	exec	Requires the protocol statement to have the value <code>tcp</code> or to be unspecified. Requires a <code>destination-port</code> value.
FTP	ftp	Requires the protocol statement to have the value <code>tcp</code> or to be unspecified. Requires a <code>destination-port</code> value.
H.323	h323	–

Table 49: Application Protocols Supported by Services Interfaces *(Continued)*

Protocol Name	CLI Value	Comments
Internet Control Message Protocol (ICMP)	icmp	Requires the protocol statement to have the value icmp or to be unspecified.
IP	ip	-
Login	login	-
NetBIOS	netbios	Requires the protocol statement to have the value udp or to be unspecified. Requires a destination-port value.
NetShow	netshow	Requires the protocol statement to have the value tcp or to be unspecified. Requires a destination-port value.
RealAudio	realaudio	-
Real-Time Streaming Protocol (RTSP)	rtsp	Requires the protocol statement to have the value tcp or to be unspecified. Requires a destination-port value.
Session Initiation Protocol	sip	-
SNMP	snmp	Requires the protocol statement to have the value udp or to be unspecified. Requires a destination-port value.
SQLNet	sqlnet	Requires the protocol statement to have the value tcp or to be unspecified. Requires a destination-port or source-port value.
Talk Program	talk	
Trace route	traceroute	Requires the protocol statement to have the value udp or to be unspecified. Requires a destination-port value.

Table 49: Application Protocols Supported by Services Interfaces (*Continued*)

Protocol Name	CLI Value	Comments
Trivial FTP (TFTP)	tftp	Requires the protocol statement to have the value <code>udp</code> or to be unspecified. Requires a destination-port value.
WinFrame	winframe	–



NOTE: You can configure application-level gateways (ALGs) for ICMP and trace route under stateful firewall, NAT, or CoS rules when twice NAT is configured in the same service set. These ALGs cannot be applied to flows created by the Packet Gateway Controller Protocol (PGCP). Twice NAT does not support any other ALGs. NAT applies only the IP address and TCP or UDP headers, but not the payload.

For more information about configuring twice NAT, see *Junos Address Aware Network Addressing Overview*.

Configuring the Network Protocol

The protocol statement allows you to specify which of the supported network protocols to match in an application definition. To configure network protocols, include the protocol statement at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]  
protocol type;
```

You specify the protocol type as a numeric value; for the more commonly used protocols, text names are also supported in the command-line interface (CLI). [Table 50 on page 523](#) shows the list of the supported protocols.

Table 50: Network Protocols Supported by Next Gen Services

Network Protocol Type	CLI Value	Comments
External Gateway Protocol (EGP)	egp	–

Table 50: Network Protocols Supported by Next Gen Services *(Continued)*

Network Protocol Type	CLI Value	Comments
Generic routing encapsulation (GR)	gre	–
ICMP	icmp	Requires an application-protocol value of icmp.
ICMPv6	icmp6	Requires an application-protocol value of icmp.
Internet Group Management Protocol (IGMP)	igmp	–
TCP	tcp	Requires a destination-port or source-port value unless you specify application-protocol rcp or dce-rcp.
UDP	udp	Requires a destination-port or source-port value unless you specify application-protocol rcp or dce-rcp.

For a complete list of possible numeric values, see RFC 1700, *Assigned Numbers (for the Internet Protocol Suite)*.



NOTE: IP version 6 (IPv6) is not supported as a network protocol in application definitions.

By default, the twice NAT feature can affect IP, TCP, and UDP headers embedded in the payload of ICMP error messages. You can include the `protocol tcp` and `protocol udp` statements with the `application` statement for twice NAT configurations. For more information about configuring twice NAT, see *Junos Address Aware Network Addressing Overview*.

Configuring the ICMP Code and Type

The ICMP code and type provide additional specification, in conjunction with the network protocol, for packet matching in an application definition. To configure ICMP settings, include the `icmp-code` and `icmp-type` statements at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]
icmp-code value;
icmp-type value;
```

You can include only one ICMP code and type value. The `application-protocol` statement must have the value `icmp`. [Table 51 on page 525](#) shows the list of supported ICMP values.

Table 51: ICMP Codes and Types Supported by Services Interfaces

CLI Statement	Description
icmp-code	<p>This value or keyword provides more specific information than <code>icmp-type</code>. Because the value's meaning depends upon the associated <code>icmp-type</code> value, you must specify <code>icmp-type</code> along with <code>icmp-code</code>. For more information, see the Routing Policies, Firewall Filters, and Traffic Policers User Guide.</p> <p>In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed). The keywords are grouped by the ICMP type with which they are associated:</p> <p>parameter-problem: ip-header-bad (0), required-option-missing (1)</p> <p>redirect: redirect-for-host (1), redirect-for-network (0), redirect-for-tos-and-host (3), redirect-for-tos-and-net (2)</p> <p>time-exceeded: ttl-eq-zero-during-reassembly (1), ttl-eq-zero-during-transit (0)</p> <p>unreachable: communication-prohibited-by-filtering (13), destination-host-prohibited (10), destination-host-unknown (7), destination-network-prohibited (9), destination-network-unknown (6), fragmentation-needed (4), host-precedence-violation (14), host-unreachable (1), host-unreachable-for-TOS (12), network-unreachable (0), network-unreachable-for-TOS (11), port-unreachable (3), precedence-cutoff-in-effect (15), protocol-unreachable (2), source-host-isolated (8), source-route-failed (5)</p>

Table 51: ICMP Codes and Types Supported by Services Interfaces (*Continued*)

CLI Statement	Description
icmp-type	<p>Normally, you specify this match in conjunction with the protocol match statement to determine which protocol is being used on the port. For more information, see the Routing Policies, Firewall Filters, and Traffic Policers User Guide.</p> <p>In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): echo-reply (0), echo-request (8), info-reply (16), info-request (15), mask-request (17), mask-reply (18), parameter-problem (12), redirect (5), router-advertisement (9), router-solicit (10), source-quench (4), time-exceeded (11), timestamp (13), timestamp-reply (14), or unreachable (3).</p>



NOTE: If you configure an interface with an input firewall filter that includes a reject action and with a service set that includes stateful firewall rules, the router executes the input firewall filter before the stateful firewall rules are run on the packet. As a result, when the Packet Forwarding Engine sends an ICMP error message out through the interface, the stateful firewall rules might drop the packet because it was not seen in the input direction.

Possible workarounds are to include a forwarding-table filter to perform the reject action, because this type of filter is executed after the stateful firewall in the input direction, or to include an output service filter to prevent the locally generated ICMP packets from going to the stateful firewall service.

Configuring Source and Destination Ports

The TCP or UDP source and destination port provide additional specification, in conjunction with the network protocol, for packet matching in an application definition. To configure ports, include the destination-port and source-port statements at the [edit applications application *application-name*] hierarchy level:

```
[edit applications application application-name]
destination-port value;
source-port value;
```

You must define one source or destination port. Normally, you specify this match in conjunction with the protocol match statement to determine which protocol is being used on the port.

You can specify either a numeric value or one of the text synonyms listed in [Table 52 on page 527](#).

Table 52: Port Names Supported by Next Gen Services

Port Name	Corresponding Port Number
snmp	161
snmptrap	162

For more information about matching criteria, see the [Routing Policies, Firewall Filters, and Traffic Policers User Guide](#).

Configuring the Inactivity Timeout Period

You can specify a timeout period for application inactivity. If the software has not detected any activity during the duration, the flow becomes invalid when the timer expires. To configure a timeout period, include the `inactivity-timeout` statement at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]
inactivity-timeout seconds;
```

The default value is 14,400 seconds. The value you configure for an application overrides any global value configured at the `[edit interfaces interface-name service-options]` hierarchy level; for more information, see *Configuring Default Timeout Settings for Services Interfaces*.

Configuring SIP

The Session Initiation Protocol (SIP) is a generalized protocol for communication between endpoints involved in Internet services such as telephony, fax, video conferencing, instant messaging, and file exchange.

The Junos OS provides ALG services in accordance with the standard described in RFC 3261, *SIP: Session Initiation Protocol*. SIP flows under the Junos OS are as described in RFC 3665, *Session Initiation Protocol (SIP) Basic Call Flow Examples*.



NOTE: Before implementing the Junos OS SIP ALG, you should be familiar with certain limitations, discussed in ["Junos OS SIP ALG Limitations" on page 535](#)

The use of NAT in conjunction with the SIP ALG results in changes in SIP header fields due to address translation. For an explanation of these translations, refer to ["SIP ALG Interaction with Network Address Translation" on page 529](#).

To implement SIP on adaptive services interfaces, you configure the `application-protocol` statement at the `[edit applications application application-name]` hierarchy level with the value `sip`. In addition, there are two other statements you can configure to modify how SIP is implemented:

- You can enable the router to accept any incoming SIP calls for the endpoint devices that are behind the NAT firewall. When a device behind the firewall registers with the proxy that is outside the firewall, the AS or Multiservices PIC maintains the registration state. When the `learn-sip-register` statement is enabled, the router can use this information to accept inbound calls. If this statement is not configured, no inbound calls are accepted; only the devices behind the firewall can call devices outside the firewall.

To configure SIP registration, include the `learn-sip-register` statement at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]  
learn-sip-register;
```



NOTE: The `learn-sip-register` statement is not applicable to the Next Gen Services MX-SPC3.

You can also manually inspect the SIP register by issuing the `show services stateful-firewall sip-register` command; for more information, see the *Junos OS System Basics and Services Command Reference*. The `show services stateful-firewall sip-register` command is not supported for Next Gen Services.

- You can specify a timeout period for the duration of SIP calls that are placed on hold. When a call is put on hold, there is no activity and flows might time out after the configured `inactivity-timeout` period expires, resulting in call state teardown. To avoid this, when a call is put on hold, the flow timer is reset to the `sip-call-hold-timeout` cycle to preserve the call state and flows for longer than the `inactivity-timeout` period.



NOTE: The `sip-call-hold-timeout` statement is not applicable to the Next Gen Services MX-SPC3.

To configure a timeout period, include the `sip-call-hold-timeout` statement at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]
sip-call-hold-timeout seconds;
```

The default value is 7200 seconds and the range is from 0 through 36,000 seconds (10 hours).

SIP ALG Interaction with Network Address Translation

The Network Address Translation (NAT) protocol enables multiple hosts in a private subnet to share a single public IP address to access the Internet. For outgoing traffic, NAT replaces the private IP address of the host in the private subnet with the public IP address. For incoming traffic, the public IP address is converted back into the private address, and the message is routed to the appropriate host in the private subnet.

Using NAT with the Session Initiation Protocol (SIP) service is more complicated because SIP messages contain IP addresses in the SIP headers as well as in the SIP body. When using NAT with the SIP service, the SIP headers contain information about the caller and the receiver, and the device translates this information to hide it from the outside network. The SIP body contains the Session Description Protocol (SDP) information, which includes IP addresses and port numbers for transmission of the media. The device translates SDP information for allocating resources to send and receive the media.

How IP addresses and port numbers in SIP messages are replaced depends on the direction of the message. For an outgoing message, the private IP address and port number of the client are replaced with the public IP address and port number of the Juniper Networks firewall. For an incoming message, the public address of the firewall is replaced with the private address of the client.

When an INVITE message is sent out across the firewall, the SIP Application Layer Gateway (ALG) collects information from the message header into a call table, which it uses to forward subsequent messages to the correct endpoint. When a new message arrives, for example an ACK or 200 OK, the ALG compares the "From:", "To:", and "Call-ID:" fields against the call table to identify the call context of the message. If a new INVITE message arrives that matches the existing call, the ALG processes it as a REINVITE.

When a message containing SDP information arrives, the ALG allocates ports and creates a NAT mapping between them and the ports in the SDP. Because the SDP requires sequential ports for the Real-Time Transport Protocol (RTP) and Real-Time Control Protocol (RTCP) channels, the ALG provides consecutive even-odd ports. If it is unable to find a pair of ports, it discards the SIP message.

This topic contains the following sections:

Outgoing Calls

When a SIP call is initiated with a SIP request message from the internal to the external network, NAT replaces the IP addresses and port numbers in the SDP and binds the IP addresses and port numbers to the Juniper Networks firewall. Via, Contact, Route, and Record-Route SIP header fields, if present, are also bound to the firewall IP address. The ALG stores these mappings for use in retransmissions and for SIP response messages.

The SIP ALG then opens pinholes in the firewall to allow media through the device on the dynamically assigned ports negotiated based on information in the SDP and the Via, Contact, and Record-Route header fields. The pinholes also allow incoming packets to reach the Contact, Via, and Record-Route IP addresses and ports. When processing return traffic, the ALG inserts the original Contact, Via, Route, and Record-Route SIP fields back into packets.

Incoming Calls

Incoming calls are initiated from the public network to public static NAT addresses or to interface IP addresses on the device. Static NATs are statically configured IP addresses that point to internal hosts; interface IP addresses are dynamically recorded by the ALG as it monitors REGISTER messages sent by internal hosts to the SIP registrar. When the device receives an incoming SIP packet, it sets up a session and forwards the payload of the packet to the SIP ALG.

The ALG examines the SIP request message (initially an INVITE) and, based on information in the SDP, opens gates for outgoing media. When a 200 OK response message arrives, the SIP ALG performs NAT on the IP addresses and ports and opens pinholes in the outbound direction. (The opened gates have a short time-to-live, and they time out if a 200 OK response message is not received quickly.)

When a 200 OK response arrives, the SIP proxy examines the SDP information and reads the IP addresses and port numbers for each media session. The SIP ALG on the device performs NAT on the addresses and port numbers, opens pinholes for outbound traffic, and refreshes the timeout for gates in the inbound direction.

When the ACK arrives for the 200 OK, it also passes through the SIP ALG. If the message contains SDP information, the SIP ALG ensures that the IP addresses and port numbers are not changed from the previous INVITE—if they are, the ALG deletes old pinholes and creates new pinholes to allow media to pass through. The ALG also monitors the Via, Contact, and Record-Route SIP fields and opens new pinholes if it determines that these fields have changed.

Forwarded Calls

A forwarded call is when, for example, user A outside the network calls user B inside the network, and user B forwards the call to user C outside the network. The SIP ALG processes the INVITE from user A as a normal incoming call. But when the ALG examines the forwarded call from B to C outside the

network and notices that B and C are reached using the same interface, it does not open pinholes in the firewall, because media will flow directly between user A and user C.

Call Termination

The BYE message terminates a call. When the device receives a BYE message, it translates the header fields just as it does for any other message. But because a BYE message must be acknowledged by the receiver with a 200 OK, the ALG delays call teardown for five seconds to allow time for transmission of the 200 OK.

Call Re-INVITE Messages

Re-INVITE messages add new media sessions to a call and remove existing media sessions. When new media sessions are added to a call, new pinholes are opened in the firewall and new address bindings are created. The process is identical to the original call setup. When one or more media sessions are removed from a call, pinholes are closed and bindings released just as with a BYE message.

Call Session Timers

The SIP ALG uses the Session-Expires value to time out a session if a Re-INVITE or UPDATE message is not received. The ALG gets the Session-Expires value, if present, from the 200 OK response to the INVITE and uses this value for signaling timeout. If the ALG receives another INVITE before the session times out, it resets all timeout values to this new INVITE or to default values, and the process is repeated.

As a precautionary measure, the SIP ALG uses hard timeout values to set the maximum amount of time a call can exist. This ensures that the device is protected should one of the following events occur:

- End systems crash during a call and a BYE message is not received.
- Malicious users never send a BYE in an attempt to attack a SIP ALG.
- Poor implementations of SIP proxy fail to process Record-Route and never send a BYE message.
- Network failures prevent a BYE message from being received.

Call Cancellation

Either party can cancel a call by sending a CANCEL message. Upon receiving a CANCEL message, the SIP ALG closes pinholes through the firewall—if any have been opened—and releases address bindings. Before releasing the resources, the ALG delays the control channel age-out for approximately five seconds to allow time for the final 200 OK to pass through. The call is terminated when the five second timeout expires, regardless of whether a 487 or non-200 response arrives.

Forking

Forking enables a SIP proxy to send a single INVITE message to multiple destinations simultaneously. When the multiple 200 OK response messages arrive for the single call, the SIP ALG parses but updates call information with the first 200 OK messages it receives.

SIP Messages

The SIP message format consists of a SIP header section and the SIP body. In request messages, the first line of the header section is the request line, which includes the method type, request-URI, and protocol version. In response messages, the first line is the status line, which contains a status code. SIP headers contain IP addresses and port numbers used for signaling. The SIP body, separated from the header section by a blank line, is reserved for session description information, which is optional. Junos OS currently supports the SDP only. The SIP body contains IP addresses and port numbers used to transport the media.

SIP Headers

In the following sample SIP request message, NAT replaces the IP addresses in the header fields to hide them from the outside network.

```
INVITE bob@10.150.20.5 SIP/2.0
Via: SIP/2.0/UDP 10.150.20.3:5434
From: alice@10.150.20.3
To: bob@10.150.20.5
Call-ID: a12abcde@10.150.20.3
Contact: alice@10.150.20.3:5434
Route: <sip:netscreen@10.150.20.3:5060>
Record-Route: <sip:netscreen@10.150.20.3:5060>
```

How IP address translation is performed depends on the type and direction of the message. A message can be any of the following:

- Inbound request
- Outbound response
- Outbound request
- Inbound response

[Table 53 on page 533](#) shows how NAT is performed in each of these cases. Note that for several of the header fields the ALG determine more than just whether the messages comes from inside or outside the

network. It must also determine what client initiated the call, and whether the message is a request or response.

Table 53: Requesting Messages with NAT Table

Inbound Request (from public to private)	To:	Replace domain with local address
	From:	None
	Call-ID:	None
	Via:	None
	Request-URI:	Replace ALG address with local address
	Contact:	None
	Record-Route:	None
	Route:	None
Outbound Response (from private to public)	To:	Replace ALG address with local address
	From:	None
	Call-ID:	None
	Via:	None
	Request-URI:	N/A
	Contact:	Replace local address with ALG address
	Record-Route:	Replace local address with ALG address

	Route:	None
Outbound Request (from private to public)	To:	None
	From:	Replace local address with ALG address
	Call-ID:	None
	Via:	Replace local address with ALG address
	Request-URI:	None
	Contact:	Replace local address with ALG address
	Record-Route:	Replace local address with ALG address
	Route:	Replace ALG address with local address
Outbound Response (from public to private)	To:	None
	From:	Replace ALG address with local address
	Call-ID:	None
	Via:	Replace ALG address with local address
	Request-URI:	N/A
	Contact:	None
	Record-Route:	Replace ALG address with local address
	Route:	Replace ALG address with local address

SIP Body

The SDP information in the SIP body includes IP addresses the ALG uses to create channels for the media stream. Translation of the SDP section also allocates resources, that is, port numbers to send and receive the media.

The following excerpt from a sample SDP section shows the fields that are translated for resource allocation.

```
o=user 2344234 55234434 IN IP4 10.150.20.3
c=IN IP4 10.150.20.3
m=audio 43249 RTP/AVP 0
```

SIP messages can contain more than one media stream. The concept is similar to attaching multiple files to an e-mail message. For example, an INVITE message sent from a SIP client to a SIP server might have the following fields:

```
c=IN IP4 10.123.33.4
m=audio 33445 RTP/AVP 0
c=IN IP4 10.123.33.4
m=audio 33447 RTP/AVP 0
c=IN IP4 10.123.33.4
m=audio 33449 RTP/AVP 0
```

Junos OS supports up to 6 SDP channels negotiated for each direction, for a total of 12 channels per call.

Junos OS SIP ALG Limitations

The following limitations apply to configuration of the SIP ALG:

- Only the methods described in RFC 3261 are supported.
- Only SIP version 2 is supported.
- TCP is not supported as a transport mechanism for signaling messages for MS-MPCs but is supported for Next Gen Services.
- *Do not configure the SIP ALG when using STUN.* if clients use STUN/TURN to detect the firewall or NAT devices between the caller and responder or proxy, the client attempts to best-guess the NAT device behavior and act accordingly to place the call.

- On MS-MPCs, do not use the endpoint-independent mapping NAT pool option in conjunction with the SIP ALG. Errors will result. This does not apply to Next Gen Services.
- IPv6 signaling data is not supported for MS-MPCs but is supported for Next Gen Services.
- Authentication is not supported.
- Encrypted messages are not supported.
- SIP fragmentation is not supported for MS-MPCs but is supported for Next Gen Services.
- The maximum UDP packet size containing a SIP message is assumed to be 9 KB. SIP messages larger than this are not supported.
- The maximum number of media channels in a SIP message is assumed to be six.
- Fully qualified domain names (FQDNs) are not supported in critical fields.
- QoS is not supported. SIP supports DSCP rewrites.
- High availability is not supported, except for warm standby.
- A timeout setting of never is not supported on SIP or NAT.
- Multicast (forking proxy) is not supported.

Configuring an SNMP Command for Packet Matching

You can specify an SNMP command setting for packet matching. To configure SNMP, include the `snmp-command` statement at the `[edit applications application application-name]` hierarchy level:

```
[edit applications application application-name]
snmp-command value;
```

The supported values are `get`, `get-next`, `set`, and `trap`. You can configure only one value for matching. The `application-protocol` statement at the `[edit applications application application-name]` hierarchy level must have the value `snmp`.

RELATED DOCUMENTATION

| *ALGs Available for Junos OS Address Aware NAT*

Examples: Configuring Application Protocols

The following example shows an application protocol definition describing a special FTP application running on port 78:

```
[edit applications]
application my-ftp-app {
    application-protocol ftp;
    protocol tcp;
    destination-port 78;
    timeout 100; # inactivity timeout for FTP service
}
```

The following example shows a special ICMP protocol (application-protocol icmp) of type 8 (ICMP echo):

```
[edit applications]
application icmp-app {
    application-protocol icmp;
    protocol icmp;
    icmp-type icmp-echo;
}
```

The following example shows a possible application set:

```
[edit applications]
application-set basic {
    http;
    ftp;
    telnet;
    nfs;
    icmp;
}
```

The software includes a predefined set of well-known application protocols. The set includes applications for which the TCP and UDP destination ports are already recognized by stateless firewall filters.

Verifying the Output of ALG Sessions

IN THIS SECTION

- [FTP Example | 538](#)
- [RTSP ALG Example | 544](#)
- [System Log Messages | 547](#)

This section contains examples of successful output from ALG sessions and information on system log configuration. You can compare the results of your sessions to check whether the configurations are functioning correctly.

FTP Example

This example analyzes the output during an active FTP session. It consists of four different flows; two are control flows and two are data flows. The example consists of the following parts:

Sample Output

MS-MPC Card

For MS-MPCs, the following is a complete sample output from the `show services stateful-firewall conversations application-protocol ftp` command:

```
user@host>show services stateful-firewall conversations application-protocol ftp
Interface: ms-1/3/0, Service set: CLBJI1-AAF001
Conversation: ALG protocol: ftp
  Number of initiators: 2, Number of responders: 2
Flow      State  Dir      Frm count
TCP       1.1.79.2:14083 ->      2.2.2.2:21  Watch  I      13
  NAT source      1.1.79.2:14083 ->    194.250.1.237:50118
TCP       1.1.79.2:14104 ->      2.2.2.2:20  Forward I      3
  NAT source      1.1.79.2:14104 ->    194.250.1.237:50119
TCP       2.2.2.2:21 ->    194.250.1.237:50118 Watch  O      12
  NAT dest      194.250.1.237:50118 ->      1.1.79.2:14083
```

```
TCP          2.2.2.2:20    -> 194.250.1.237:50119 Forward  0          5
NAT dest     194.250.1.237:50119  ->      1.1.79.2:14104
```

For each flow, the first line shows flow information, including protocol (TCP), source address, source port, destination address, destination port, flow state, direction, and frame count.

- The state of a flow can be Watch, Forward, or Drop:
 - A Watch flow state indicates that the control flow is monitored by the ALG for information in the payload. NAT processing is performed on the header and payload as needed.
 - A Forward flow forwards the packets without monitoring the payload. NAT is performed on the header as needed.
 - A Drop flow drops any packet that matches the 5 tuple.
- The frame count (Frm count) shows the number of packets that were processed on that flow.

The second line shows the NAT information.

- source indicates source NAT.
- dest indicates destination NAT.
- The first address and port in the NAT line are the original address and port being translated for that flow.
- The second address and port in the NAT line are the translated address and port for that flow.

MX-SPC3 Card

On the MX-SPC3 services card, the following is a complete sample output from the `show services sessions application-protocol ftp` operational mode command:

```
user@host>show services sessions application-protocol ftp
Session ID: 536870917, Service-set: ss1, Policy name: p1/131085, Timeout: 1, Valid
Logical system: root-logical-system
Resource information : FTP ALG, 1, 1
  In: 12.10.10.10/35281 --> 22.20.20.3/8204;tcp, Conn Tag: 0x0, If: vms-2/0/0.100, Pkts: 6,
Bytes: 320,
  Out: 22.20.20.3/8204 --> 60.1.1.2/48747;tcp, Conn Tag: 0x0, If: vms-2/0/0.200, Pkts: 9, Bytes:
8239,

Session ID: 536870919, Service-set: ss1, Policy name: p1/131085, Timeout: 29, Valid
Logical system: root-logical-system
```



```
Resource information : FTP ALG, 1, 0
  In: 12.10.10.10/44194 --> 22.20.20.3/21;tcp, Conn Tag: 0x0, If: vms-2/0/0.100, Pkts: 13,
Bytes: 585,
  Out: 22.20.20.3/21 --> 60.1.1.2/48660;tcp, Conn Tag: 0x0, If: vms-2/0/0.200, Pkts: 11, Bytes:
650,
Total sessions: 2
```

For each session:

- The first line shows flow information, including session ID, service-set name, policy name, session timeout, logical system name, and its state.
- The second line, Resource information, indicates the session is created by ALG, including the ALG name (FTP ALG) and ASL group id, which is 1 and the ASL resource id, which is 0 for control session and 1 for data session.
- The third line In is forward flow and the fourth line Out is reverse flow, including the source address, source port, destination address, destination port, protocol (TCP), session conn-tag, incoming for In and outgoing for Out interface, received frame count and bytes. NAT is performed on the header as needed.

FTP System Log Messages

System log messages are generated during an FTP session. For more information about system logs, see ["System Log Messages" on page 547](#).

MS-MPC Card

The following system log messages are generated during creation of the FTP control flow:

- Rule Accept system log:

```
Oct 27 11:42:54 (FPC Slot 1, PIC Slot 1) {ss_ftp}[FWNAT]: ASP_SFW_RULE_ACCEPT: proto 6 (TCP)
application: ftp, fe-3/3/3.0:1.1.1.2:4450 -> 2.2.2.2:21, Match SFW accept rule-set:, rule:
ftp, term: 1
```

- Create Accept Flow system log:

```
Oct 27 11:42:54 (FPC Slot 1, PIC Slot 1) {ss_ftp}[FWNAT]: ASP_SFW_CREATE_ACCEPT_FLOW: proto 6
(TCP) application: ftp, fe-3/3/3.0:1.1.1.2:4450 -> 2.2.2.2:21, creating forward or watch flow
```

- System log for data flow creation:

```
Oct 27 11:43:30 (FPC Slot 1, PIC Slot 1) {ss_ftp}[FWNAT]: ASP_SFW_FTP_ACTIVE_ACCEPT: proto 6
(TCP) application: ftp, so-2/1/2.0:2.2.2.2:20 -> 1.1.1.2:50726, Creating FTP active mode
forward flow
```

MX-SPC3 CardCard

The following system log messages are generated during creation of the FTP control flow:

- System log for FTP control session creation:

```
Mar 23 23:58:54 esst480r RT_FLOW: RT_FLOW_SESSION_CREATE_USF: Tag svc-set-name ss1: session
created 20.1.1.2/52877->30.1.1.2/21 0x0 junos-ftp 20.1.1.2/52877->30.1.1.2/21 0x0 N/A N/A N/A
N/A 6 p1 ss1-ZoneIn ss1-ZoneOut 818413576 N/A(N/A) ge-1/0/2.0 UNKNOWN UNKNOWN UNKNOWN N/A N/A
-1 N/A
```

```
Mar 23 23:59:00 esst480r junos-alg: RT_ALG_FTP_ACTIVE_ACCEPT: application:ftp data,
vms-3/0/0.0 30.1.1.2:20 -> 20.1.1.2:33947 (TCP)
```

- System log for FTP data session creation:

```
Mar 23 23:59:00 esst480r RT_FLOW: RT_FLOW_SESSION_CREATE_USF: Tag svc-set-name ss1: session
created 30.1.1.2/20->20.1.1.2/33947 0x0 junos-ftp-data 30.1.1.2/20->20.1.1.2/33947 0x0 N/A
N/A N/A N/A 6 p1 ss1-ZoneOut ss1-ZoneIn 818413577 N/A(N/A) ge-1/1/6.0 FTP-DATA UNKNOWN
UNKNOWN Infrastructure File-Servers 2 N/A
```

- System log for FTP data session destroy:

```
Mar 23 23:59:02 esst480r RT_FLOW: RT_FLOW_SESSION_CLOSE_USF: Tag svc-set-name ss1: session
closed TCP FIN: 30.1.1.2/20->20.1.1.2/33947 0x0 junos-ftp-data 30.1.1.2/20->20.1.1.2/33947
0x0 N/A N/A N/A N/A 6 p1 ss1-ZoneOut ss1-ZoneIn 818413577 2954(4423509) 281(14620) 2 FTP-DATA
UNKNOWN N/A(N/A) ge-1/1/6.0 No Infrastructure File-Servers 2 N/A
```

- System log for FTP control session destroy:

```
Mar 23 23:59:39 esst480r RT_FLOW: RT_FLOW_SESSION_CLOSE_USF: Tag svc-set-name ss1: session
closed Closed by junos-tcp-clt-emul: 20.1.1.2/52877->30.1.1.2/21 0x0 junos-ftp 20.1.1.2/52877-
>30.1.1.2/21 0x0 N/A N/A N/A N/A 6 p1 ss1-ZoneIn ss1-ZoneOut 818413576 23(1082) 18(1176) 45
UNKNOWN UNKNOWN N/A(N/A) ge-1/0/2.0 No N/A N/A -1 N/A
```

Analysis

Control Flows

MS-MPC Card

The control flows are established after the three-way handshake is complete.

- Control flow from FTP client to FTP server. TCP destination port is 21.

```
TCP          1.1.79.2:14083 ->      2.2.2.2:21    Watch    I          13
NAT source   1.1.79.2:14083  ->    194.250.1.237:50118
```

- Control flow from FTP server to FTP client. TCP source port is 21.

```
TCP          2.2.2.2:21    ->    194.250.1.237:50118 Watch    0          12
NAT dest     194.250.1.237:50118 ->      1.1.79.2:14083
```

MX-SPC3 Card

The control flows are established after the three-way handshake is complete.

- Control session from FTP client to FTP server, TCP destination port is 21.

```
Session ID: 536870919, Service-set: ss1, Policy name: p1/131085, Timeout: 29, Valid
Logical system: root-logical-system
Resource information : FTP ALG, 1, 0
  In: 12.10.10.10/44194 --> 22.20.20.3/21;tcp, Conn Tag: 0x0, If: vms-2/0/0.100, Pkts: 13,
  Bytes: 585,
  Out: 22.20.20.3/21 --> 60.1.1.2/48660;tcp, Conn Tag: 0x0, If: vms-2/0/0.200, Pkts: 11,
  Bytes: 650,
```

- Data session from FTP client to FTP server, it's for FTP passive mode.

```
Session ID: 536870917, Service-set: ss1, Policy name: p1/131085, Timeout: 1, Valid
Logical system: root-logical-system
Resource information : FTP ALG, 1, 1
  In: 12.10.10.10/35281 --> 22.20.20.3/8204;tcp, Conn Tag: 0x0, If: vms-2/0/0.100, Pkts: 6,
  Bytes: 320,
  Out: 22.20.20.3/8204 --> 60.1.1.2/48747;tcp, Conn Tag: 0x0, If: vms-2/0/0.200, Pkts: 9,
  Bytes: 8239,
```

- Data session from FTP server to FTP client, it's for FTP active mode:

```
Session ID: 549978117, Service-set: ss1, Policy name: p1/131085, Timeout: 1, Valid
Logical system: root-logical-system
Resource information : FTP ALG, 1, 1
  In: 22.20.20.3/20 --> 60.1.1.3/6049;tcp, Conn Tag: 0x0, If: vms-2/0/0.200, Pkts: 10, Bytes:
  8291,
  Out: 12.10.10.10/33203 --> 22.20.20.3/20;tcp, Conn Tag: 0x0, If: vms-2/0/0.100, Pkts: 5,
  Bytes: 268,
```

Data Flows

A data port of 20 is negotiated for data transfer during the course of the FTP control protocol. These two flows are data flows between the FTP client and the FTP server:

TCP	1.1.79.2:14104 ->	2.2.2.2:20	Forward I	3
NAT source	1.1.79.2:14104	-> 194.250.1.237:50119		
TCP	2.2.2.2:20 ->	194.250.1.237:50119	Forward 0	5
NAT dest	194.250.1.237:50119	-> 1.1.79.2:14104		

Troubleshooting Questions

1. How do I know if the FTP ALG is active?

- The ALG protocol field in the conversation should display ftp.
- There should be a valid frame count (Frm count) in the control flows.
- A valid frame count in the data flows indicates that data transfer has taken place.

2. What do I need to check if the FTP connection is established but data transfer does not take place?

- Most probably, the control connection is up, but the data connection is down.
- Check the conversations output to determine whether both the control and data flows are present.

3. How do I interpret each flow? What does each flow mean?

- FTP control flow initiator flow—Flow with destination port 21
- FTP control flow responder flow—Flow with source port ;21
- FTP data flow initiator flow—Flow with destination port 20
- FTP data flow responder flow—Flow with source port 20

RTSP ALG Example

The following is an example of an RTSP conversation. The application uses the RTSP protocol for control connection. Once the connection is set up, the media is sent using UDP protocol (RTP).

This example consists of the following:

Sample Output for MS-MPCs

Here is the output from the `show services stateful-firewall conversations operational mode` command:

```
user@host# show services stateful-firewall conversations
Interface: ms-3/2/0, Service set: svc_set
Conversation: ALG protocol: rtsp
  Number of initiators: 5, Number of responders: 5
```

Flow	State	Dir	Frm	count
TCP	1.1.1.3:58795	->	2.2.2.2:554	Watch I 7
UDP	1.1.1.3:1028	->	2.2.2.2:1028	Forward I 0
UDP	1.1.1.3:1029	->	2.2.2.2:1029	Forward I 0
UDP	1.1.1.3:1030	->	2.2.2.2:1030	Forward I 0
UDP	1.1.1.3:1031	->	2.2.2.2:1031	Forward I 0
TCP	2.2.2.2:554	->	1.1.1.3:58795	Watch 0 5
UDP	2.2.2.2:1028	->	1.1.1.3:1028	Forward 0 6
UDP	2.2.2.2:1029	->	1.1.1.3:1029	Forward 0 0
UDP	2.2.2.2:1030	->	1.1.1.3:1030	Forward 0 3
UDP	2.2.2.2:1031	->	1.1.1.3:1031	Forward 0 0

Sample Output for MX-SPC3 Services Card

Here is the output from the `show services sessions application-protocol rtsp operational mode` command:

```

user@host# run show services sessions application-protocol rtsp
Session ID: 1073741828, Service-set: sset1, Policy name: p1/131081, Timeout: 116, Valid
Logical system: root-logical-system
Resource information : RTSP ALG, 1, 0
  In: 31.0.0.2/33575 --> 41.0.0.2/554;tcp, Conn Tag: 0x0, If: vms-4/0/0.1, Pkts: 8, Bytes: 948,
  Out: 41.0.0.2/554 --> 131.10.0.1/7777;tcp, Conn Tag: 0x0, If: vms-4/0/0.2, Pkts: 6, Bytes:
1117,

Session ID: 1073741829, Service-set: sset1, Policy name: p1/131081, Timeout: 120, Valid
Logical system: root-logical-system
Resource information : RTSP ALG, 1, 1
  In: 41.0.0.2/35004 --> 131.10.0.1/7780;udp, Conn Tag: 0x0, If: vms-4/0/0.2, Pkts: 220, Bytes:
79200,
  Out: 31.0.0.2/30004 --> 41.0.0.2/35004;udp, Conn Tag: 0x0, If: vms-4/0/0.1, Pkts: 0, Bytes: 0,

Session ID: 1073741830, Service-set: sset1, Policy name: p1/131081, Timeout: 120, Valid
Logical system: root-logical-system
Resource information : RTSP ALG, 1, 4
  In: 41.0.0.2/35006 --> 131.10.0.1/7781;udp, Conn Tag: 0x0, If: vms-4/0/0.2, Pkts: 220, Bytes:
174240,
  Out: 31.0.0.2/30006 --> 41.0.0.2/35006;udp, Conn Tag: 0x0, If: vms-4/0/0.1, Pkts: 0, Bytes: 0,
Total sessions: 3

```

Analysis

An RTSP conversation should consist of TCP flows corresponding to the RTSP control connection. There should be two flows, one in each direction, from client to server and from server to client:

TCP	1.1.1.3:58795 ->	2.2.2.2:554	Watch	I	7
TCP	2.2.2.2:554 ->	1.1.1.3:58795	Watch	0	5

- The RTSP control connection for the initiator flow is sent from destination port 554.
- The RTSP control connection for the responder flow is sent from source port 554.

The UDP flows correspond to RTP media sent over the RTSP connection.

Troubleshooting Questions

1. Media does not work when the RTSP ALG is configured. What do I do?

- Check RTSP conversations to see whether both TCP and UDP flows exist.
- The ALG protocol should be displayed as rtsp.



NOTE: The state of the flow is displayed as `Watch`, because the ALG processing is taking place and the client is essentially “watching” or processing payload corresponding to the application. For FTP and RTSP ALG flows, the control connections are always `Watch` flows.

2. How do I check for ALG errors?

- You can check for errors by issuing the following command. Each ALG has a separate field for ALG packet errors.

```
user@host# show services stateful-firewall statistics extensive
Interface: ms-3/2/0
Service set: svc_set
New flows:
  Accepts: 1347, Discards: 0, Rejects: 0
Existing flows:
  Accepts: 144187, Discards: 0, Rejects: 0
Drops:
  IP option: 0, TCP SYN defense: 0
  NAT ports exhausted: 0
Errors:
  IP: 0, TCP: 276
  UDP: 0, ICMP: 0
  Non-IP packets: 0, ALG: 0
IP errors:
  IP packet length inconsistencies: 0
  Minimum IP header length check failures: 0
  Reassembled packet exceeds maximum IP length: 0
  Illegal source address: 0
  Illegal destination address: 0
  TTL zero errors: 0, Illegal IP protocol number (0 or 255): 0
  Land attack: 0
  Non-IPv4 packets: 0, Bad checksum: 0
  Illegal IP fragment length: 0
```

```

IP fragment overlap: 0
IP fragment reassembly timeout: 0
Unknown: 0
TCP errors:
  TCP header length inconsistencies: 0
  Source or destination port number is zero: 0
  Illegal sequence number and flags combinations: 0
  SYN attack (multiple SYN messages seen for the same flow): 276
  First packet not a SYN message: 0
  TCP port scan (TCP handshake, RST seen from server for SYN): 0
  Bad SYN cookie response: 0
UDP errors:
  IP data length less than minimum UDP header length (8 bytes): 0
  Source or destination port number is zero: 0
  UDP port scan (ICMP error seen for UDP flow): 0
ICMP errors:
  IP data length less than minimum ICMP header length (8 bytes): 0
  ICMP error length inconsistencies: 0
  Duplicate ping sequence number: 0
  Mismatched ping sequence number: 0
ALG errors:
  BOOTP: 0, DCE-RPC: 0, DCE-RPC portmap: 0
  DNS: 0, Exec: 0, FTP: 0
  ICMP: 0
  Login: 0, NetBIOS: 0, NetShow: 0
  RPC: 0, RPC portmap: 0
  RTSP: 0, Shell: 0
  SNMP: 0, SQLNet: 0, TFTP: 0
  Traceroute: 0

```

System Log Messages

Enabling system log generation and checking the system log are also helpful for ALG flow analysis. This section contains the following:

System Log Configuration

You can configure the enabling of system log messages at a number of different levels in the Junos OS CLI. As shown in the following sample configurations, the choice of level depends on how specific you want the event logging to be and what options you want to include. For details on the configuration options, see the [Junos OS Administration Library for Routing Devices](#) (system level) or the [Junos OS Services Interfaces Library for Routing Devices](#) (all other levels).

1. At the topmost global level:

```
user@host# show system syslog
file messages {
    any any;
}
```

2. At the service set level:

```
user@host# show services service-set svc_set
syslog {
    host local {
        services any;
    }
}
stateful-firewall-rules allow_rtsp;
interface-service {
    service-interface ms-3/2/0;
}
```

3. At the service rule level:

```
user@host# show services stateful-firewall rule allow_rtsp
match-direction input-output;
term 0 {
    from {
        applications junos-rtsp;
    }
    then {
        accept;
        syslog;
    }
}
```

System Log Output

System log messages are generated during flow creation, as shown in the following examples:

The following system log message indicates that the ASP matched an accept rule:

```
Oct 25 16:11:37 (FPC Slot 3, PIC Slot 2) {svc_set}[FWNAT]: ASP_SFW_RULE_ACCEPT: proto 6 (TCP)  
application: rtsp, ge-2/0/1.0:1.1.1.2:35595 -> 2.2.2.2:554, Match SFW accept rule-set: , rule:  
allow_rtsp, term: 0
```

For a complete listing of system log messages, see the [System Log Explorer](#).

12

PART

NAT, Stateful Firewall, and IDS Flows

- [Inline NAT Services Overview and Configuration | 551](#)
-

Inline NAT Services Overview and Configuration

IN THIS CHAPTER

- [Inline Static Source NAT Overview | 551](#)
- [Configuring Inline Static Source NAT44 for Next Gen Services | 552](#)
- [Inline Static Destination NAT Overview | 556](#)
- [Configuring Inline Static Destination NAT for Next Gen Services | 556](#)
- [Inline Twice Static NAT Overview | 560](#)
- [Configuring Inline Twice Static NAT44 for Next Gen Services | 561](#)

Inline Static Source NAT Overview

IN THIS SECTION

- [Benefits | 552](#)

Inline static source NAT uses the capabilities of the MPC line card to perform address translation, eliminating the need for a services card.

Static source NAT performs a one-to-one static mapping of the original private domain host source address to a public source address. A block of external addresses is set aside for this mapping, and source addresses are translated as hosts in a private domain originate sessions to the external domain. Static source NAT does not perform port mapping. For packets outbound from the private network, static source NAT translates source IP addresses and related fields such as IP, TCP, UDP, and ICMP header checksums. For inbound packets, static source NAT translates the destination IP address and the checksums.

Benefits

- Allows hosts in the private network to connect with the external domain, while hiding the private network.
- Eliminates the need for a services card
- Supports more NAT flows than a services card

Configuring Inline Static Source NAT44 for Next Gen Services

IN THIS SECTION

- [Configuring the Source Pool for Inline Static Source NAT44 | 552](#)
- [Configuring the NAT Rule for Inline Static Source NAT44 | 553](#)
- [Configuring the Service Set for Inline Static Source NAT44 | 554](#)
- [Configuring Inline Services and an Inline Services Interface | 555](#)

Configuring the Source Pool for Inline Static Source NAT44

To configure the source pool for inline static source NAT44:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]  
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix to address address-prefix
```

3. Configure a one-to-one static mapping of the original source addresses to the addresses in the source pool by specifying the first address from the matching source-address prefix that is in the source NAT rule.

```
[edit services nat source pool nat-pool-name]
user@host# set host-address-base ip-address
```

4. To allow the IP addresses of a NAT source pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rule for Inline Static Source NAT44

To configure the NAT source rule for inline static source NAT44:

1. Configure the NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

4. Specify the NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

5. Configure the generation of a syslog when traffic matches the NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Inline Static Source NAT44

To configure the service set for inline static source NAT44:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service set, which requires a single service interface, or a next-hop service set, which requires an inside and outside service interface.

- To configure an interface service set:

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface si-slot-number/pic-number/0.logical-unit-number
```

- To configure a next-hop service set:

```
[edit services service-set service-set-name]
[edit services service-set service-set-name]
```

```
user@host# set next-hop-service inside-service-interface vms-slot-number/pic-number/
0.logical-unit-number outside-service-interface si-slot-number/pic-number/0.logical-unit-
number
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Configuring Inline Services and an Inline Services Interface

To enable inline services and an inline services interface:

1. Enable inline services for the FPC and PIC slot, and define the amount of bandwidth to dedicate to inline services.

```
[edit chassis si-fpc slot-number pic number]
user@host# set inline-services bandwidth (1g | 10g | 20g | 30g | 40g | 100g)
```

2. Configure the inline services logical interface or interfaces.

- If you are using an interface service set, configure one logical unit:

```
[edit interfaces si-slot-number/pic-number/0]
user@host# set unit logical-unit-number family family
```

- If you are using a next-hop service set, configure two logical units and define the inside and outside interfaces:

```
[edit interfaces si-slot-number/pic-number/0]
user@host# set unit logical-unit-number family family
user@host# set unit logical-unit-number service-domain inside
user@host# set unit logical-unit-number family family
user@host# set unit logical-unit-number service-domain outside
```


Inline Static Destination NAT Overview

IN THIS SECTION

- [Benefits | 556](#)

Inline static destination NAT uses the capabilities of the MPC line card to perform address translation, eliminating the need for a services card.

Static destination NAT translates the IPv4 destination address of an incoming packet to the IPv4 address of a private server. This redirects traffic destined to a virtual host (identified by the original destination IP address) to the real host (identified by the translated destination IP address).

Static destination NAT uses a one-to-one mapping between the original address and the translated address; the mapping is configured statically.

Benefits

- Allows external traffic to communicate with a private host without revealing the host's private IP address
- Does not require port mapping
- Eliminates the need for a services card
- Supports more NAT flows than a services card

Configuring Inline Static Destination NAT for Next Gen Services

IN THIS SECTION

- [Configuring the Destination Pool for Inline Static Destination NAT | 557](#)
- [Configuring the NAT Rule for Inline Static Destination NAT | 557](#)
- [Configuring the Service Set for Inline Static Destination NAT | 559](#)
- [Configuring Inline Services and an Inline Services Interface | 559](#)

Configuring the Destination Pool for Inline Static Destination NAT

To configure the destination pool for inline static destination NAT:

1. Create a destination pool.

```
user@host# edit services nat destination pool nat-pool-name
```

2. Define the addresses or subnets to which destination addresses are translated.

```
[edit services nat destination pool nat-pool-name]
user@host# set address address-prefix
```

3. To allow the IP addresses of a NAT destination pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rule for Inline Static Destination NAT

To configure the NAT destination for static destination NAT:

1. Configure the NAT rule name.

```
[edit services destination source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out)
```

3. Specify the source addresses of traffic that the NAT rule applies to.

To specify one address or prefix value:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

4. Specify the destination addresses that the NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

5. Specify the NAT pool that contains the destination addresses for translated traffic.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat pool nat-pool-name
```

6. Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Inline Static Destination NAT

To configure the service set for inline static destination NAT:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service set, which requires a single service interface, or a next-hop service set, which requires an inside and outside service interface.

- To configure an interface service set:

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface si-slot-number/pic-number/0.logical-unit-number
```

- To configure a next-hop service set:

```
[edit services service-set service-set-name]
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface si-slot-number/pic-number/0.logical-unit-number outside-service-interface si-slot-number/pic-number/0.logical-unit-number
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Configuring Inline Services and an Inline Services Interface

To enable inline services and an inline services interface:

1. Enable inline services for the FPC and PIC slot, and define the amount of bandwidth to dedicate to inline services.

```
[edit chassis si-fpc slot-number pic number port number]
user@host# set inline-services bandwidth (1g | 10g | 20g | 30g | 40g | 100g)
```

2. Configure the inline services logical interface or interfaces.

- If you are using an interface service set, configure one logical unit:

```
[edit interfaces si-slot-number/pic-number/0
user@host# set unit logical-unit-number family family
```

- If you are using a next-hop service set, configure two logical units and define the inside and outside interfaces:

```
[edit interfaces si-slot-number/pic-number/0
user@host# set unit logical-unit-number family family
user@host# set unit logical-unit-number service-domain inside
user@host# set unit logical-unit-number family family
user@host# set unit logical-unit-number service-domain outside
```

Inline Twice Static NAT Overview

IN THIS SECTION

- [Benefits | 561](#)

Inline twice static NAT uses the capabilities of the MPC line card to perform address translation, eliminating the need for a services card.

Twice static NAT translates both the source and destination IP addresses. An addresses is translated with a one-to-one static mapping to an address in a pool. Port mapping is not performed.

The original private domain host source address is translated to a public source address.

The destination address is translated to the IPv4 address of a private server. This redirects traffic destined to a virtual host (identified by the original destination IP address) to the real host (identified by the translated destination IP address).

Benefits

- Allows hosts in the private network to connect with the external domain, while hiding the private network.
- Hides a private network
- Allows external traffic to communicate with a private host without revealing the host's private IP address
- Does not require port mapping
- Eliminates the need for a services card
- Supports more NAT flows than a services card

Configuring Inline Twice Static NAT44 for Next Gen Services

IN THIS SECTION

- [Configuring the Source and Destination Pools for Inline Twice Static NAT44 | 561](#)
- [Configuring the NAT Rules for Inline Twice Static NAT44 | 562](#)
- [Configuring the Service Set for Inline Twice Static NAT44 | 564](#)
- [Configuring Inline Services and an Inline Services Interface | 565](#)

Configuring the Source and Destination Pools for Inline Twice Static NAT44

To configure the source and destination pools for inline twice static NAT44:

1. Create a source pool.

```
user@host# edit services nat source pool nat-pool-name
```

2. Define the addresses or subnets to which source addresses are translated.

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix
```

or

```
[edit services nat source pool nat-pool-name]
user@host# set address address-prefix to address address-prefix
```

3. Configure a one-to-one static mapping of the original source addresses to the addresses in the source pool by specifying the first address from the matching source-address prefix that is in the source NAT rule.

```
[edit services nat source pool nat-pool-name]
user@host# set host-address-base ip-address
```

4. Create a destination pool. Do not use the same name that you used for the source pool.

```
user@host# edit services nat destination pool nat-pool-name
```

5. Define the addresses or subnets to which destination addresses are translated.

```
[edit services nat destination pool nat-pool-name]
user@host# set address address-prefix
```

6. To allow the IP addresses of a NAT pool to overlap with IP addresses in pools used in other service sets, configure `allow-overlapping-pools`.

```
[edit services nat]
user@host# set allow-overlapping-pools
```

Configuring the NAT Rules for Inline Twice Static NAT44

To configure the source and destination NAT rules for twice static NAT44:

1. Configure the source NAT rule name.

```
[edit services nat source]
user@host# set rule-set rule-set-name rule rule-name
```

2. Specify the traffic direction to which the source NAT rule set applies.

```
[edit services nat source rule-set rule-set-name]
user@host# set match-direction (in | out)
```

3. Specify the addresses that are translated by the source NAT rule.

To specify one address or prefix value:

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set match source-address-name address-name
```

4. Specify the source NAT pool that contains the addresses for translated traffic.

```
[edit services nat source rule-set rule-set-name rule rule-name]
user@host# set then source-nat pool nat-pool-name
```

5. Configure the generation of a syslog when traffic matches the source NAT rule conditions.

```
[edit services nat source rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

6. Configure the destination NAT rule name.

```
[edit services nat destination]
user@host# set rule-set rule-set-name rule rule-name
```


7. Specify the traffic direction to which the destination NAT rule set applies.

```
[edit services nat destination rule-set rule-set-name]
user@host# set match-direction (in | out | in-out)
```

8. Specify the destination addresses of traffic that the destination NAT rule applies to.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address address
```

To specify a range of addresses, configure an address book global address with the desired address range, and assign the global address to the NAT rule:

```
[edit services address-book global]
user@host# set address address-name range-address lower-limit to upper-limit
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address-name address-name
```

To specify any unicast address:

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set match destination-address any-unicast
```

9. Specify the destination NAT pool that contains the destination addresses for translated traffic.

```
[edit services nat destination rule-set rule-set-name rule rule-name]
user@host# set then destination-nat pool nat-pool-name
```

10. Configure the generation of a syslog when traffic matches the destination NAT rule match conditions.

```
[edit services nat destination rule-set rule-set-name rule rule-name then]
user@host# set syslog
```

Configuring the Service Set for Inline Twice Static NAT44

To configure the service set for inline static NAT44:

1. Define the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

2. Configure either an interface service set, which requires a single service interface, or a next-hop service set, which requires an inside and outside service interface.

- To configure an interface service set:

```
[edit services service-set service-set-name]
user@host# set interface-service service-interface si-slot-number/pic-number/0.logical-unit-number
```

- To configure a next-hop service set:

```
[edit services service-set service-set-name]
[edit services service-set service-set-name]
user@host# set next-hop-service inside-service-interface si-slot-number/pic-number/0.logical-unit-number
outside-service-interface vms-slot-number/pic-number/0.logical-unit-number
```

3. Specify the NAT rule sets to be used with the service set.

```
[edit services service-set service-set-name]
user@host# set nat-rule-sets rule-set-name
```

Configuring Inline Services and an Inline Services Interface

To enable inline services and an inline services interface:

1. Enable inline services for the FPC and PIC slot, and define the amount of bandwidth to dedicate to inline services.

```
[edit chassis fpc slot-number pic number]
user@host# set inline-services bandwidth (1g | 10g | 20g | 30g | 40g | 100g)
```

2. Configure the inline services logical interface or interfaces.

- If you are using an interface service set, configure one logical unit:

```
[edit interfaces si-slot-number/pic-number/0]
user@host# set unit logical-unit-number family family
```

- If you are using a next-hop service set, configure two logical units and define the inside and outside interfaces:

```
[edit interfaces si-slot-number/pic-number/0]
user@host# set unit logical-unit-number family family
user@host# set unit logical-unit-number service-domain inside
user@host# set unit logical-unit-number family family
user@host# set unit logical-unit-number service-domain outside
```

13

PART

Configuration Statements

- [Junos CLI Reference Overview | 568](#)
-

Junos CLI Reference Overview

We've consolidated all Junos CLI commands and configuration statements in one place. Read this guide to learn about the syntax and options that make up the statements and commands. Also understand the contexts in which you'll use these CLI elements in your network configurations and operations.

- [Junos CLI Reference](#)

Click the links to access Junos OS and Junos OS Evolved configuration statement and command summary topics.

- [Configuration Statements](#)
- [Operational Commands](#)