Softwire Services for Juniper Service Framework (JSF)

Published: 2012-02-28

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

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

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

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

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

Supported Platforms

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

- M Series
- T Series
- MX Series
- J Series

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

For more information about the **load** command, see the *Junos OS CLI User Guide*.

Documentation Conventions

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

Table 1: Notice Icons

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

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

Table 2: Text and Syntax Conventions

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

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

Convention	Description	Examples
Italic text like this	• Introduces important new terms. • Identifies book names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS System Basics Configuration Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; interface names; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Enclose optional keywords or variables.	**stub** <**default-metric** *metric*>**;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** (***string1*** **|** ***string2*** **|** ***string3***)
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Enclose a variable for which you can substitute one or more values.	**community name members [** ***community-ids*** **]**

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

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

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

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

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

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://www.juniper.net/alerts/

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

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

Opening a Case with JTAC

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

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

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

PART 1

Overview

CHAPTER 1

Software Services

Tunneling Services for IPv4-to-IPv6 Transition Overview

The Junos OS enables service providers to transition to IPv6 by using softwire encapsulation and decapsulation techniques. A softwire is a tunnel that is created between softwire Customer Premises Equipment (CPE). A softwire 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 softwire, unlike a typical mesh of generic routing encapsulation (GRE) tunnels that would require you to do so. A softwire initiator at the customer end encapsulates native packets and tunnels them to a softwire concentrator at the service provider. The softwire concentrator decapsulates the packets and sends them to their destination. A softwire is created when a softwire concentrator receives the first tunneled packet of a flow and prepares for flow processing. The softwire exists as long as the softwire concentrator is providing flows for routing. A flow counter is maintained; when the number of active flows is 0, the softwire is deleted. Statistics are kept for both flows and softwires.

Softwire addresses are not specifically configured under any physical or virtual interface. Therefore, the number of established softwires does not affect throughput, and scalability is independent of the number of interfaces. The scalability is only limited to the number of flows that the platform (services DPC or PIC) can support.

This topic contains the following sections:

6to4 Overview

Basic 6to4

6to4 is an Internet transition mechanism for migrating from IPv4 to IPv6, a system that allows IPv6 packets to be transmitted over an IPv4 network (generally the IPv4 Internet) without the need to configure explicit tunnels. 6to4 is described in RFC 3056, *Connection of IPv6 Domains via IPv4 Clouds*. 6to4 is especially relevant during the initial phases of deployment to full, native IPv6 connectivity, since IPv6 is not required on nodes between the host and the destination. However, it is intended only as a transition mechanism and is not meant to be used permanently.

6to4 can be used by an individual host, or by a local IPv6 network. When used by a host, it must have a global IPv4 address connected, and the host is responsible for the encapsulation of outgoing IPv6 packets and the decapsulation of incoming 6to4 packets. If the host is configured to forward packets for other clients, often a local network, it is then a router.

There are two kinds of 6to4 virtual routers: border routers and relay routers. A 6to4 border router is an IPv6 router supporting a 6to4 pseudointerface, and is normally the border router between an IPv6 site and a wide-area IPv4 network. A relay router is a 6to4 router configured to support transit routing between 6to4 addresses and pure native IPv6 addresses.

In order for a 6to4 host to communicate with the native IPv6 Internet, its IPv6 default gateway must be set to a 6to4 address which contains the IPv4 address of a 6to4 relay router. To avoid the need for users to set this up manually, the Anycast address of 192.88.99.1 has been allocated to send packets to a 6to4 relay router. Note that when wrapped in 6to4 with the subnet and hosts fields set to zero, this IPv4 address (192.88.99.1) becomes the IPv6 address 2002:c058:6301::. To ensure BGP routing propagation, a short prefix of 192.88.99.0/24 has been allocated for routes pointed at 6to4 relay routers that use this Anycast IP address. Providers willing to provide 6to4 service to their clients or peers should advertise the Anycast prefix like any other IP prefix, and route the prefix to their 6to4 relay.

Packets from the IPv6 Internet to 6to4 systems must be sent to a 6to4 relay router by normal IPv6 routing methods. The specification states that such relay routers must only advertise 2002::/16 and not subdivisions of it to prevent IPv4 routes from polluting the routing tables of IPv6 routers. From there they can then be sent over the IPv4 Internet to the destination.

6to4 Anycast

Router 6to4 assumes that 6to4 routers and relays are managed and configured cooperatively. In particular, 6to4 sites must configure a relay router to carry the outbound traffic, which becomes the default IPv6 router (except for 2002::/16). The objective of the Anycast variant, defined in RFC 3068, *An Anycast Prefix for 6to4 Relay Routers*, is to avoid the need for such configuration. This makes the solution available for small or domestic users, even those with a single host or simple home gateway instead of a border router. This is achieved by defining 192.88.99.1 as the default IPv4 address for a 6to4 relay, and 2002:c058:6301:: as the default IPv6 router prefix (“well-known prefix”) for a 6to4 site.

RFC 6343, *Advisory Guidelines for 6to4 Deployment*, published in August 2011, identifies a wide range of problems associated with the use of unmanaged 6to4 Anycast relay routers.

6to4 Provider-Managed Tunnels

A solution to many problems associated with unmanaged Anycast 6to4 is presented in IETF informational draft draft-kuarsingh-v6ops-6to4-provider-managed-tunnel-02, *6to 4 Provider-Managed Tunnels (PMT)*. That document, a "work in progress," proposes a solution that allows providers to exercise greater control over the routing of 6to4 traffic.

Anycast 6to4 implies a default configuration for the user site. It does not require any particular user action. It does require an IPv4 Anycast route to be in place to a relay at 192.88.99.1. Traffic does not necessarily return to the same 6to4 gateway because of the the "well-known" 6to4 prefix used and advertised by all 6to4 traffic.

6to4 provider-managed tunnels (PMTs) facilitate the management of 6to4 tunnels using an Anycast configuration. 6to4 PMT enables service providers to improve 6to4 operation when network conditions provide suboptimal performance or break normal 6to4 operation. 6to4 PMT provides a stable provider prefix and forwarding environment by utilizing existing 6to4 relays with an added function of IPv6 prefix translation that controls the flow of return traffic.

The 6to4 managed tunnel model behaves like a standard 6to4 service between the customer IPv6 host or gateway and the 6to4 PMT relay (within the provider domain). The 6to4-PMT relay shares properties with 6rd (RFC5969) by decapsulating and forwarding embedded IPv6 flows, within an IPv4 packet, to the IPv6 Internet. The model provides an additional function which translates the source 6to4 prefix to a provider assigned prefix which is not found in 6rd (RFC5969) or traditional 6to4 operation. The 6to4-PMT relay provides a stateless (or stateful) mapping of the 6to4 prefix to a provider-supplied prefix by mapping the embedded IPv4 address in the 6to4 prefix to the provider prefix.

DS-Lite Softwires—IPv4 over IPv6

When an Internet service provider (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 (CE) WAN equipment to reach the IPv4 network. DS-Lite enables IPv4 customers to continue to access the Internet using their current hardware by using a softwire initiator, 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.

DS-Lite is supported on Multiservices 100, 400, and 500 PICs on M Series routers and on MX Series routers equipped with Multiservices Dense Port Concentrator (DPCs).

NOTE: IPv6 Provider Edge (6PE), or MPLS-enabled IPv6, is available for ISPs with MPLS-enabled networks. These networks now can use Multiprotocol Border Gateway Protocol (MP-BGP) to provide connectivity between the DS-Lite B4 and AFTR (or any two IPv6 nodes). DS-Lite properly handles encapsulation and decapsulation despite the presence of additional MPLS header information.

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:

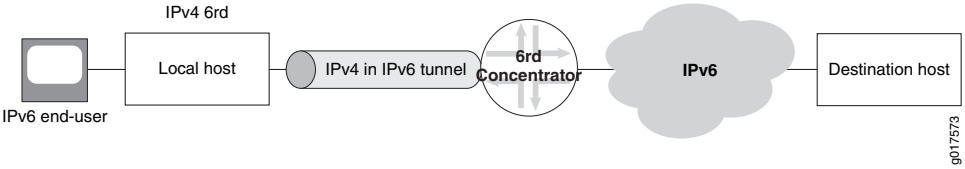
- The term *softwire initiator* has been replaced by *B4*.
- The term *softwire concentrator* has been replaced by *AFTR*.

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.

6rd Softwires—IPv6 over IPv4

6rd softwire flow is shown in Figure 1 on page 6.

Figure 1: 6rd Softwire Flow



The Junos OS supports a 6rd softwire concentrator on a services DPC or PIC to facilitate rapid deployment of IPv6 service to subscribers on native IPv4 CE WANs. IPv6 packets are encapsulated in IPv4 packets by a softwire initiator at the CE WAN. These packets are tunneled to a softwire concentrator residing on a multiservices DPC (branch relay). 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. All of these functions are performed in a single pass of the services PIC.

In the reverse path, IPv6 packets are sent to the Services DPC where they are encapsulated in IPv4 packets corresponding to the proper softwire and sent to the CE WAN.

The softwire concentrator creates softwires as the IPv4 packets are received from the CE WAN side or IPV6 packets are received from the Internet. A 6rd softwire on the Services DPC is identified by the 3-tuple containing the service set ID, CE softwire initiator IPv4 address, and softwire concentrator IPv4 address. IPv6 flows are also created for the

encapsulated IPv6 payload, and are associated with the specific softwire that carried them in the first place. When the last IPv6 flow associated with a softwire ends, the softwire is deleted. This simplifies configuration and there is no need to create or manage tunnel interfaces.

6rd is supported on Multiservices 100, 400, and 500 PICs on M Series and T Series routers, and on MX Series platforms equipped with Multiservices DPCs.

For more information on 6rd softwires, see RFC 5969, *IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification*.

Related Documentation

- See Network Address Translation Overview.

PART 2

Configuration

CHAPTER 2

Configuration Tasks

Configuring a 6rd Softwire Concentrator

To configure a 6rd softwire concentrator:

1. Assign a name to the 6rd softwire concentrator.

   ```
   [edit services softwire softwire-concentrator]
   user@host# edit v6rd v6rd-softwire-concentator
   ```

2. Specify the address of the softwire tunnel.

   ```
   [edit services softwire softwire-concentrator v6rd v6rd-softwire-concentator]
   user@host# set softwire-address address
   ```

3. Specify the MTU for the softwire tunnel.

   ```
   [edit services softwire softwire-concentrator v6rd v6rd-softwire-concentator]
   user@host# set mtu-v4 mtu-v4
   ```

TIP: In this release there is no support for fragmentation and reassembly, therefore the MTUs on the IPv6 and IPV4 network must be properly configured by the administrator.

Configuring a DS-Lite Softwire Concentrator

To configure a DS-Lite softwire concentrator:

1. Assign a name to the DS-Lite softwire concentrator.

   ```
   [edit services softwire softwire-concentrator]
   user@host# edit ds-lite ds-lite-softwire-concentrator
   ```

2. Specify the address of the softwire tunnel.

   [edit services softwire softwire-concentrator ds-lite *ds-lite-softwire-concentrator*]
   user@host# **set softwire-address** *address*

3. Specify the MTU for the softwire tunnel.

   [edit services softwire softwire-concentrator ds-lite *ds-lite-softwire-concentrator*]
   user@host# **set mtu-v6** *mtu-v6*

   NOTE: This option sets the maximum transmission unit when encapsulating IPv4 packets into IPv6. If the final length is greater than the MTU, the IPv6 packet will be fragmented. This option is mandatory since it depends on other network parameters under administrator control.

4. To copy DSCP information from the IPv6 header into the decapsulated IPv4 header, include the **copy-dscp** statement.

   [edit services softwire softwire-concentrator ds-lite *ds-lite-softwire-concentrator*]
   user@host# **set copy-dscp**

5. Specify the maximum number of flows for the softwire:

   [edit services softwire softwire-concentrator ds-lite *ds-lite-softwire-concentrator*]
   user@host# **set flow-limit 1000**

Configuring Softwire Rules

You configure softwire rules to instruct the router how to direct traffic to the addresses specified for 6rd or DS-Lite softwire concentrators. Softwire 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 6rd or DS-Lite softwire concentrator.

You can create a softwire rule consisting of one or more terms and associate a particular 6rd or DS-Lite softwire concentrator with each term. You can include the softwire rule in service sets along with other services rules.

To configure a softwire rule:

1. Assign a name to the rule.

   [edit services softwire]
   user@host# **edit** rule *rule-name*

2. Specify the match direction.

   [edit services softwire rule *rule-name*]
   user@host# **set match-direction** (input | output)

3. Assign a name for the first term.

   [edit services softwire rule *rule-name*]
   user@host# **edit term** *term-name*

4. Associate a 6rd or DS-Lite softwire concentrator with this term.

   [edit services softwire rule *rule-name* term *term-name*]

```
user@host# set then ds-lite name
```

or

```
user@host# set then v6rd v6rd-softwire-concentator
```

5. Repeat Steps 3 and 4 for as many additional terms as needed.

Configuring Stateful Firewall Rules for 6rd Softwire

You must configure a stateful firewall rule for use with 6rd softwires. The stateful firewall service is used only to direct packets to the softwire, not for firewalling purposes. The 6rd softwire service itself must be stateless. To support stateless processing, you must include an **allow** term in both directions of the stateful firewall policy.

To include a stateful firewall rule for 6rd softwire processing:

1. Assign a name to the rule.

```
[edit services stateful-firewall]
user@host# edit rule rule-name
```

2. Specify the match direction.

```
[edit services stateful-firewall rule-name]
user@host# set match-direction input-output
```

3. Assign a name for the term.

```
[edit services stateful-firewall rule-name]
user@host# edit term term-name
```

4. Specify that all traffic in both directions should be accepted for the softwire process.

```
[edit services stateful-firewall rule-name term term-name]
user@host# set then accept
```

Configuring IPv6 Multicast Interfaces

Configure multicast filters on Ethernet interfaces when IPv6 NAT is used for neighbor discovery. This enables the router to process softwire-initiated flows in both directions.

To configure IPv6 multicast interfaces:

1. Access the softwire hierarchy.

```
user@host# edit services softwire
```

2. Include the **ipv6-multicast-interfaces** statement for an individual interface.

```
[edit services softwire]
user@host# set ipv6-multicast-interfaces interface-name
```

Or configure all softwire interfaces as IPv6 multicast.

```
[edit services softwire]
user@host# set ipv6-multicast-interfaces all
```

Configuring Service Sets for Softwire

You must include softwire rules or a softwire rule set in a service set to enable softwire processing. You must include a stateful firewall rule for DS-Lite.

To configure service sets for softwire:

1. Include a softwire rule or rule set in the service set.

   [edit services service-set *service-set-name*]
   user@host# **set** softwire-rules **rule** ***softwire-rule-name***

2. When using a 6rd softwire, include a stateful-firewall rule.

   [edit services service-set *service-set-name*]
   user@host# **set stateful-firewall-rules*softwire-rule-name***

3. 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 softwire 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.

For further information, see Configuring Service Rules.

CHAPTER 3

Example

Examples: Configuring 6rd Softwires

This topic consists of the following sections:

Configuring a 6rd Concentrator for IPv6 Internet Connectivity

This example describes how a 6rd concentrator can be configured for a 6rd domain, D1, to provide IPv6 Internet connectivity.

1. Configure the softwire concentrator and softwire rule.

```
[edit services]
softwire-concentrator {
  v6rd v6rd-dom1 {
    softwire-address 30.30.30.1;
    ipv4-prefix 10.10.10.0/24;
    v6rd-prefix 3040::0/16;
    mtu-v4 9192;
  }
}
  rule v6rd-dom1-r1 {
    match-direction input;
    term t1 {
      then {
        v6rd v6rd-dom1;
      }
    }
  }
```

   Here, softwire-address 30.30.30.1 is the softwire concentrator IPv4 address, 10.10.10.0/24 is the IPv4 prefix of the CE WAN side, and 3040::0/16 is the IPv6 prefix of the 6rd domain D1.

2. Define a stateful firewall rule.

```
[edit services stateful-firewall]
rule r1 {
  match-direction input-output;
  term t1 {
    then {
      accept;
    }
  }
}
```

   You must configure a stateful-firewall rule that accepts all traffic in both the input and output direction in order for 6rd to work; however, this is not enforced through the CLI. This is because in IPv6, gratuitous IPv6 packets are expected (due to anycast) and should not be dropped. The service PIC can handle reverse traffic without seeing forward traffic all. This can also happen in the case of service PIC switchover in the middle of a session. By default, the stateful firewall on the service PIC will drop all traffic unless a rule is configured explicitly to allow it.

Configuring 6rd and DS-Lite in the Same Service Set

This example describes how 6rd can be configured with DS-Lite on the same PIC and in the same service set.

1. Configure the ingress interface.

```
[edit interfaces]
  ge-1/2/0 {
    unit 0 {
      family inet {
        service {
          input {
            service-set v6rd-dslite-service-set;
          }
          output {
            service-set v6rd-dslite-service-set;
          }
        }
        address 10.10.10.1/24;
      }
      family inet6 {
        service {
          input {
            service-set v6rd-dslite-service-set;
          }
          output {
            service-set v6rd-dslite-service-set;
          }
        }
        address 2001::1/16;
      }
    }
  }
```

   Here the service set is applied on the inet (IPv4) and inet6 (IPv6) families of subunit 0. Both DS-Lite IPv6 traffic and 6rd IPv4 traffic hits the service filter and is sent to the services PIC.

2. Configure the egress interface (IPv6 internet).

```
[edit interfaces]
  ge-1/2/2 {
    unit 0 {
      family inet {
        address 200.200.200.1/24;
      }
      family inet6 {
        address 3ABC::1/16;
      }
    }
  }
```

   The IPv4 server that the DS-Lite clients are trying to reach is at 200.200.200.2/24 and the IPv6 server is at 3ABC::2/16.

3. Configure the services PIC.

```
[edit interfaces}
  sp-3/0/0 {
    unit 0 {
```



```
        family inet;
        family inet6;
      }
    }
```

4. Configure the service set.

```
[edit services]
  service-set v6rd-dslite-service-set {
    softwire-rules v6rd-r1;
    softwire-rules dslite-r1;
    stateful-firewall-rules r1;
    nat-rules dslite-nat-r1;
    interface-service {
      service-interface sp-3/0/0;
    }
  }
```

   This service set has a stateful firewall rule and 6rd rule for 6rd service. The service set also includes a softwire rule for DS-Lite and a NAT rule to perform address translation for all the DS-Lite traffic. The NAT rule performs NAPT translation in the forward direction on the source address and port of the DS-Lite traffic.

5. Configure the softwire concentrators.

```
[edit services softwire-concentrators]
  softwire-concentrator {
    ds-lite ds1 {
      softwire-address 1001::1;
      mtu-v6 9192;
    }
    v6rd v6rd-dom1 {
      softwire-address 30.30.30.1;
      ipv4-prefix 10.10.10.0/24;
      v6rd-prefix 3040::0/16;
      mtu-v4 9192;
    }
  }
```

6. Edit the softwire rules.

```
[edit services softwire]
  rule v6rd-r1 {
    match-direction input;
    term t1 {
      then {
        v6rd v6rd-dom1;
      }
    }
  }
  rule v6rd-r1 {
    match-direction input;
    term t1 {
      then {
        v6rd v6rd-dom1;
      }
    }
  }
  rule dslite-r1 {
  match-direction input;
    term dslite-t1 {
      then {
        ds-lite ds1;
      }
    }
```

The following routes are added by the services pic daemon on the Routing Engine.

```
user@router# run show route 30.30.30.1

inet.0: 43 destinations, 46 routes (42 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

30.30.30.1/32      *[Static/786432] 00:24:11
                       Service to v6rd-dslite-service-set

[edit]
user@router# run show route 3040::0/16

inet6.0: 23 destinations, 33 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

3040::/16          *[Static/786432] 00:24:39
                       Service to v6rd-dslite-service-set

user@router# run show route 1001::1

inet6.0: 33 destinations, 43 routes (33 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1001::1/128        *[Static/1] 1w2d 22:05:41
                       Service to v6rd-dslite-service-set
```

7. Configure a stateful firewall rule.

```
[edit services stateful-firewall]
  rule r1 {
    match-direction input-output;
    term t1 {
      then {
        accept;
      }
    }
  }
```

8. Configure NAT for DS-Lite.

```
[edit services nat]
  pool dslite-pool {
    address-range low 33.33.33.1 high 33.33.33.32;
    port {
      automatic;
    }
  }
    rule dslite-nat-r1 {
      match-direction input;
      term dslite-nat-t1 {
        from {
          source-address {
            20.20.0.0/16;
          }
        }
        then {
          translated {
            source-pool dslite-pool;
            translation-type {
              source dynamic;
            }
          }
        }
      }
    }
```

Because of this NAT rule, the following NAT routes are installed for the reverse DS-Lite traffic.

```
user@router# user@router# run show route 33.33.33.0/24 show route 33.33.33.0/24

inet.0: 48 destinations, 52 routes (47 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

33.33.33.1/32      *[Static/1] 1w2d 23:08:38
                      Service to v6rd-dslite-service-set
33.33.33.2/31      *[Static/1] 1w2d 23:08:38
                      Service to v6rd-dslite-service-set
33.33.33.4/30      *[Static/1] 1w2d 23:08:38
                      Service to v6rd-dslite-service-set
33.33.33.8/29      *[Static/1] 1w2d 23:08:38
                      Service to v6rd-dslite-service-set
33.33.33.16/28     *[Static/1] 1w2d 23:08:38
                      Service to v6rd-dslite-service-set
33.33.33.32/32     *[Static/1] 1w2d 23:08:38
                      Service to v6rd-dslite-service-set

The NAT rule will do NAPT the traffic coming from 20.20.0.0/16 to public
 address range 33.33.33.1 to 33.33.33.32
```

CHAPTER 4

Configuration Statements

ds-lite

Syntax

```
ds-lite ds-lite-softwire-concentrator{
    auto-update-mtu;
    copy-dscp;
    flow-limit flow-limit;
    mtu-v6 mtu-v6;
    softwire-address softwire-address;
  }
}
```

Hierarchy Level [edit services softwire softwire-concentrator]

Release Information

Statement introduced in Junos OS Release 10.4.
auto-update-mtu option introduced in Junos OS Release 10.4.
copy-dscp option introduced in Junos OS Release 11.2.
mtu-v6 option introduced in Junos OS Release 10.4.
softwire-address option introduced in Junos OS Release 10.4.

Description Configure settings for a DS-Lite concentrator used to process IPv4 packets encapsulated in IPv6.

Options

ds-lite-softwire-concentrator—Name applied to a DS-Lite softwire concentrator.

auto-update-mtu—This option is not currently supported.

copy-dscp—Copy DSCP information to IPv4 headers during decapsultation.

flow-limit—Maximum number of IPv4 flows per softwire (0 through 16384).

mtu-v6—Maximum transmission unit (MTU), in bytes (0 through 9192), for encapsulating IPv4 packets into IPv6. If the final length is greater than the configured value, the IPv6 packet is fragmented.

softwire-address—Address of the DS-Lite softwire concentrator.

Required Privilege Level

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

Related Documentation

- Softwire Configuration Guidelines
- Configuring a DS-Lite Softwire Concentrator on page 11

rule (Softwire)

Syntax

```
rule rule-name {
  match-direction (input | output);
  term term-name {
    then {
      (ds-lite ds-lite-softwire-concentrator | v6rd v6rd-softwire-concentrator);
    }
  }
}
```

Hierarchy Level [edit services softwire],
[edit services softwire rule-set *rule-set-name*]

Release Information Statement introduced in Junos OS Release 10.4.

Description Configure a rule to apply a softwire concentrator for a flow.

Options ***rule-name***—Identifier for the collection of terms that constitute this rule.

input—Apply the rule match on the input side of the interface.

output—Apply the rule match on the output side of the interface.

The remaining statements are explained separately.

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

Related Documentation
- Configuring Softwire Rules on page 12

rule-set (Softwire)

Syntax

```
rule-set rule-set-name {
  rule rule-name;
}
```

Hierarchy Level [edit services softwire]

Release Information Statement introduced in Junos OS Release 10.4.

Description Specify the rule set the router uses when applying this service.

Options ***rule-set-name***—Identifier for the collection of rules that constitute this rule set.

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

Related Documentation
- Configuring Softwire Rules on page 12

softwire-concentrator

Syntax

```
softtwire-concentrator {
  ds-lite ds-lite-softwire-concentrator {
    auto-update-mtu;
    flow-limit flow-limit;
    mtu-v6 mtu-v6;
    softwire-address address;
  }
  v6rd v6rd-softwire-concentator {
    ipv4-prefix ipv4-prefix;
    v6rd-prefix ipv6-prefix;
    mtu-v4 mtu-v4;
  }
}
```

Hierarchy Level [edit services softwire]

Release Information Statement introduced in Junos OS Release 10.4.

Description Configure settings for a softwire concentrator.

Options The statements are explained separately.

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

Related Documentation

- Softwire Configuration Guidelines

softwire-rules

Syntax (softwire-rule *rule-name* | softwire-rule-sets *rule-set-name*);

Hierarchy Level [edit services service-set *service-set-name*]

Release Information Statement introduced in Junos OS Release 10.4.

Description Specify the DS-Lite or 6rd rules or rule set included in this service set. You can configure multiple rules; however, you can only configure one rule set for each service set.

Options ***rule-name***—Identifier for the collection of terms that constitute this rule.

rule-set-name—Identifier for the set of rules to be included.

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

Related Documentation

- Configuring Service Rules

v6rd

Syntax

```
v6rd v6rd-softwire-concentator {
  ipv4-prefix ipv4-prefix;
  v6rd-prefix ipv6-prefix;
  mtu-v4 mtu-v4;
  softwire-address ipv4-address;
}
```

Hierarchy Level [edit services softwire softwire-concentrator]

Release Information Statement introduced in Junos OS Release 10.4.

Description Configure settings for a 6rd concentrator used to process IPv6 packets encapsulated in IPv4 packets.

Options

ipv4-prefix—IPv4 prefix of the customer edge (CE) network

ipv6-prefix—IPv6 prefix of the 6rd domain.

mtu-v4— Maximum transmission unit (MTU), in bytes (576 through 9192), for IPv6 packets enacapsulated into IPv4. If the final length is greater than the configured value, the IPv4 packet will be dropped.

address—IPv4 address of a softwire concentrator. This is an IPv4 address independent of any interface and on a different prefix.

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

Related Documentation

- Softwire Configuration Guidelines

PART 3

Administration

CHAPTER 5

Softwire Operational Mode Commands

clear services softwire statistics

Syntax **clear services softwire statistics**
<inferface *interface-name*>

Release Information Command introduced in Junos OS Release 10.4.

Description Clear softwire statistics.

Options **interface *interface-name*** —(Optional) Name of the interface servicing the softwire. When you omit this option, data for all interfaces are cleared.

Required Privilege Level view

List of Sample Output clear services softwire statistics on page 32

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

Sample Output

clear services softwire statistics

```
user@host> clear services softwire statistics
```

show services softwire

Syntax **show services softwire**
<count>

Release Information Command introduced in Junos OS Release 10.4.
<count> option added in Junos OS Release 11.2.

Description Display information about softwire services. Information is displayed on both 6rd and DS-Lite services.

Options **count** ***interface-name*** —(Optional) Display the current softwire counts for a service set for both DS-Lite and 6rd.

Required Privilege Level view

List of Sample Output show services softwire on page 33
show services softwire count on page 33

Output Fields Table 3 on page 33 lists the output fields for the **command-name** command. Output fields are listed in the approximate order in which they appear.

Table 3: show-services-softwire Output Fields

Field Name	Field Description	Level of Output
Interface	Interface for which information is displayed.	All levels
Service Set	Service set containing the softwire rules for the interface.	All levels
Softwire	Name of the softwire concentrator.	All levels
Direction	Direction of the flow.	All levels
Flow count	Number of flows.	All levels

Sample Output

show services softwire

```
user@host> show services softwire
Interface: sp-3/0/0, Service set: v6rd-dom1-dom3-service-set
Softwire                                    Direction     Flow count
10.10.10.2      ->     30.30.30.1              I                 13
```

show services softwire count

```
user@host> show services softwire count
Interface   Service set                  DS-Lite        6RD
sp-0/0/0    dslite-svc-set1              2              0
```

show services stateful-firewall flows

Syntax

show services stateful-firewall flows
<brief | extensive | summary | terse>
<application-protocol *protocol*>
<count>
<destination-port *destination-port*>
<destination-prefix *destination-prefix*>
<interface *interface-name*>
<limit *number*>
<protocol *protocol*>
<service-set *service-set*>
<source-port *source-port*>
<source-prefix *source-prefix*>

Release Information

Command introduced before Junos OS Release 7.4.
pgcp option introduced in Junos OS Release 8.4.
application-protocol option introduced in Junos OS Release 10.4.

Description

Display stateful firewall flow table entries. When the interface is used for softwire processing, the type of softwire concentrator (**DS-LITE** or **6rd**) is shown, and frame counts are provided.

Options

none—Display standard information about all stateful firewall flows.

brief | extensive | summary | terse—(Optional) Display the specified level of output.

application-protocol *application-protocol*—(Optional) Display information about one of the following application-level gateway (ALG) protocol types:

- **bootp**—Bootstrap protocol
- **dce-rpc**—Distributed Computing Environment (DCE) remote procedure call (RPC) protocol

NOTE: Use this option to select Microsoft Remote Procedure Call (MSRPC).

- **dce-rpc-portmap**—Distributed Computing Environment (DCE) remote procedure call (RPC) portmap protocol
- **dns**—Domain Name Service protocol
- **exec**—Remote execution protocol
- **ftp**—File Transfer Protocol
- **h323**—H.323 protocol
- **icmp**—Internet Control Message Protocol
- **iiop**—Internet Inter-ORB Protocol

- **ip**—Internet protocol
- **netbios**—NetBIOS protocol
- **netshow**—Netshow protocol
- **pptp** —Point-to-Point Tunneling Protocol
- **realaudio**—RealAudio protocol
- **rpc**—Remote Procedure Call protocol

NOTE: Use this option to select Sun Microsystems Remote Procedure Call protocol (SunRPC).

- **rpc-portmap**—Remote Procedure Call portmap protocol
- **rtsp**—Real-Time Streaming Protocol
- **sip**—Session Initiation Protocol
- **snmp**—Simple Network Management Protocol
- **talk**—Talk protocol
- **tftp**—Trivial File Transfer Protocol
- **traceroute**—Traceroute
- **winframe**—WinFrame

count—(Optional) Display a count of the matching entries.

destination-port ***destination-port***—(Optional) Display information for a particular destination port. The range of values is from 0 to 65535.

destination-prefix ***destination-prefix***—(Optional) Display information for a particular destination prefix.

interface ***interface-name***—(Optional) Display information about a particular interface. On M Series and T Series routers, ***interface-name*** can be **ms-*fpc/pic/port*** or **rsp*number***. On J Series routers, ***interface-name*** is **ms-*pim*/0/*port***.

limit ***number***—(Optional) Maximum number of entries to display.

protocol ***protocol***—(Optional) Display information about one of the following IP types:

- ***number***—Numeric protocol value from 0 to 255
- **ah**—IPsec Authentication Header protocol
- **egp**—An exterior gateway protocol
- **esp**—IPsec Encapsulating Security Payload protocol
- **gre**—A generic routing encapsulation protocol

- **icmp**—Internet Control Message Protocol
- **igmp**—Internet Group Management Protocol
- **ipip**—IP-within-IP Encapsulation Protocol
- **ospf**—Open Shortest Path First protocol
- **pim**—Protocol Independent Multicast protocol
- **rsvp**—Resource Reservation Protocol
- **sctp**—Stream Control Protocol
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

service-set *service-set*—(Optional) Display information for a particular service set.

source-port *source-port*—(Optional) Display information for a particular source port. The range of values is from 0 to 65535.

source-prefix *source-prefix*—(Optional) Display information for a particular source prefix.

Required Privilege Level view

Related Documentation
- clear services stateful-firewall flows

List of Sample Output
show services stateful-firewall flows on page 37
show services stateful-firewall flows (For Softwire Flows) on page 37
show services stateful-firewall flows brief on page 38
show services stateful-firewall flows extensive on page 38
show services stateful-firewall flows count on page 38
show services stateful-firewall flows destination port on page 38
show services stateful-firewall flows source port on page 38
show services stateful-firewall flows (Twice NAT) on page 38

Output Fields Table 4 on page 36 lists the output fields for the **show services stateful-firewall flows** command. Output fields are listed in the approximate order in which they appear.

Table 4: show services stateful-firewall flows Output Fields

Field Name	Field Description
Interface	Name of the interface.
Service set	Name of a service set. Individual empty service sets are not displayed. If no service set has any flows, a flow table header is displayed for each service set.
Flow Count	Number of flows in a session.
Flow or Flow Prot	Protocol used for this flow.

Table 4: show services stateful-firewall flows Output Fields *(continued)*

Field Name	Field Description
Source	Source prefix of the flow in the format ***source-prefix:port***. For ICMP flows, port information is not displayed.
Dest	Destination prefix of the flow. For ICMP flows, port information is not displayed.
State	Status of the flow: • **Drop**—Drop all packets in the flow without response. • **Forward**—Forward the packet in the flow without looking at it. • **Reject**—Drop all packets in the flow with response. • **Watch**—Inspect packets in the flow.
Dir	Direction of the flow: input **(I)** or output **(O)**.
Frm count	Number of frames in the flow.

Sample Output

show services stateful-firewall flows

```
user@host> show services stateful-firewall flows
Interface: ms-1/3/0, Service set: green

Flow
Prot     Source                Dest                State      Dir     Frm count
TCP     10.58.255.178:23   ->    10.59.16.100:4000 Forward    O
TCP      10.58.255.50:33005->   10.58.255.178:23   Forward    I             1
  Source NAT    10.58.255.50:33005->    10.59.16.100:4000
  Destin NAT    10.58.255.178:23   ->         0.0.0.0:4000
```

show services stateful-firewall flows (For Softwire Flows)

When a service set includes softwire processing, the following output format is used for the softwire flows:

```
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  O          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  O          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  O          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
```

show services stateful-firewall flows brief

The output for the **show services stateful-firewall flows brief** command is identical to that for the **show services stateful-firewall flows** command. For sample output, see show services stateful-firewall flows.

show services stateful-firewall flows extensive

```
user@host> show services stateful-firewall flows extensive
Interface: ms-0/3/0, Service set: ss_nat
Flow                                                  State     Dir       Frm
count
TCP           16.1.0.1:2330  ->      16.49.0.1:21        Forward    I
   8
    NAT source        16.1.0.1:2330    ->       16.41.0.1:2330
    NAT dest         16.49.0.1:21      ->       16.99.0.1:21
  Byte count: 455, TCP established, TCP window size: 57344
  TCP acknowledge: 3251737524, TCP tickle enabled, tcp_tickle: 0
  Flow role: Master, Timeout: 720
TCP          16.99.0.1:21    ->      16.41.0.1:2330      Forward    O
   5
    NAT source       16.99.0.1:21      ->       16.49.0.1:21
    NAT dest         16.41.0.1:2330    ->        16.1.0.1:2330
  Byte count: 480, TCP established, TCP window size: 57344
  TCP acknowledge: 463128048, TCP tickle enabled, tcp_tickle: 0
  Flow role: Responder, Timeout: 720
```

show services stateful-firewall flows count

```
user@host> show services stateful-firewall flows count
Interface             Service set                                  Flow Count

ms-1/3/0              green                                                 2
```

show services stateful-firewall flows destination port

```
user@router> show services stateful-firewall flows destination-port 21
Interface: ms-0/3/0, Service set: svc_set_trust
Flow
                                                 State    Dir       Frm count
Interface: ms-0/3/0, Service set: svc_set_untrust
Flow                                            State    Dir       Frm count
TCP         10.50.10.2:2143  ->     10.50.20.2:21    Watch    O               0
```

show services stateful-firewall flows source port

```
user@router> show services stateful-firewall flows source-port 2143
Interface: ms-0/3/0, Service set: svc_set_trust
Flow
                                              State    Dir       Frm count
Interface: ms-0/3/0, Service set: svc_set_untrust
Flow                                            State    Dir       Frm count
TCP         10.50.10.2:2143  ->     10.50.20.2:21    Watch    O               0
```

show services stateful-firewall flows (Twice NAT)

```
user@router> show services stateful-firewall flows
Flow                                          State    Dir       Frm count
UDP          40.0.0.8:23439 ->     80.0.0.1:16485   Watch    I             20
    NAT source        40.0.0.8:23439   ->     172.16.1.10:1028
    NAT dest          80.0.0,1:16485   ->     192.16.1.10:22415
UDP      192.16.1.10:22415  ->  172.16.1.10:1028    Watch    O             20
    NAT source     192.16.1.10:22415   ->        80.0.0.1:16485
    NAT dest       172.16.1.10:1028    ->        40.0.0.8:23439
```

PART 4

Index

Index