6RD CONFIGURATION GUIDE
MX Series/M Series Routers with MS-DPC/MS-PIC

Although Juniper Networks has attempted to provide accurate information in this guide, Juniper Networks does not warrant or guarantee the accuracy of the information provided herein. Third party product descriptions and related technical details provided in this document are for information purposes only and such products are not supported by Juniper Networks. All information provided in this guide is provided “as is”, with all faults, and without warranty of any kind, either expressed or implied or statutory. Juniper Networks and its suppliers hereby disclaim all warranties related to this guide and the information contained herein, whether expressed or implied of statutory including, without limitation, those of merchantability, fitness for a particular purpose and noninfringement, or arising from a course of dealing, usage, or trade practice.
# Table of Contents

1 Introduction ................................................................................................................. 4  
2 Scope .......................................................................................................................... 4  
3 Design Considerations ............................................................................................... 4  
4 Terminology ............................................................................................................... 4  
5 Implementation: Juniper 6rd Solution ........................................................................ 5  
6 6rd Feature Summary ................................................................................................. 5  
7 Configuration Examples ............................................................................................. 6  
   7.1 Topology ................................................................................................................ 6  
   7.2 Basic 6rd Configuration ........................................................................................ 7  
   7.3 6rd Routes ............................................................................................................ 8  
   7.4 Advertising 6rd Routes ....................................................................................... 8  
      7.4.1 Interface Style Service Set ............................................................................ 8  
      7.4.2 Next-hop Style Service Set .......................................................................... 10  
   7.5 Pinging 6rd Softwire Concentrator ..................................................................... 12  
   7.6 6rd Domain to Internet ....................................................................................... 12  
      7.6.1 Interfaces Configuration .............................................................................. 12  
      7.6.2 Service Set Configuration ............................................................................ 13  
      7.6.3 Softwire Concentrator Configuration .......................................................... 13  
      7.6.4 Stateful Firewall Configuration .................................................................. 13  
      7.6.5 Interfaces Configuration .............................................................................. 14  
      7.6.6 Service Set Configuration ............................................................................ 15  
      7.6.7 Softwire Concentrator Configuration .......................................................... 15  
      7.6.8 Stateful Firewall Configuration .................................................................. 16  
   7.7 6to4 Using 6rd ....................................................................................................... 16  
      7.7.1 Interfaces Configuration .............................................................................. 16  
      7.7.2 Service Set Configuration ............................................................................ 17  
      7.7.3 Softwire Concentrator Configuration .......................................................... 17  
      7.7.4 Stateful Firewall Configuration .................................................................. 17  
   7.8 6rd Anycast (Redundancy) ................................................................................... 18  
      7.8.1 Interfaces Configuration .............................................................................. 18  
      7.8.2 Service Set Configuration ............................................................................ 19  
      7.8.3 Softwire Concentrator Configuration .......................................................... 19  
      7.8.4 Stateful Firewall Configuration .................................................................. 21  
   7.9 6rd + DS-Lite in Same Service Set ....................................................................... 21  
      7.9.1 Interfaces Configuration .............................................................................. 21  
      7.9.2 Service Set Configuration ............................................................................ 22  
      7.9.3 Softwire Concentrator Configuration .......................................................... 23  
      7.9.4 Stateful Firewall Configuration .................................................................. 24  
      7.9.5 NAT Configuration ...................................................................................... 24
Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Figure 1: Packet flow on the BR</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Figure 2: Topology illustrating multiple 6rd domains</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Figure 3: Topology used to illustrate CLI commands for 6rd deployment and operations</td>
<td>32</td>
</tr>
</tbody>
</table>
1 Introduction
ISPs today have to address the problem of IPv4 address exhaustion and in parallel enable IPv6 services over IPv4 infrastructure. IPv6 and IPv4 are expected to coexist for a long time to come. Providers are looking at mixed approaches, from enabling IPv6 services in the least disruptive ways on legacy infrastructure to deploying IPv6 from the get-go in greenfield applications. 6rd is one mechanism to start offering IPv6 services over IPv4 infrastructure.

2 Scope
This implementation guide provides an overview of 6rd with configuration steps and basic design guidelines when implementing 6rd with the Service-PIC or MS-DPC blade on the Juniper Networks® MX Series 3D Universal Edge Routers. This guide also covers some operational aspects and coexistence of other IPv6 transition mechanisms such as DS-Lite. For an in-depth look at IPv6 transition mechanisms and methods to address IPv4 address exhaustion, please refer to http://www.juniper.net/ipv6.

3 Design Considerations
IPv6 rapid deployment (6rd) enables ISPs to deploy IPv6 unicast service to IPv4 sites rapidly. It allows ISPs to reuse the existing IPv4 access network infrastructure and provide connectivity to islands of v6 networks. The IPv6 packets from a host (laptop/PC) are carried in IPv6-in-IPv4 tunnels from the 6rd customer edge (CE) routers to the 6rd Border Relay(s) (BR). These IPv6-in-IPv4 tunnels are called softwires [RFC4925] and terminate on the BR, also called softwire concentrator (SC), at some point in the network where IPv6 packets are decapsulated and forwarded. 6rd builds on 6to4 tunneling concepts and overcomes some of its limitations. The key difference with 6to4 is that 6rd addresses are derived from an IPv6 prefix tied to the service provider address space, guaranteeing return reachability of the IPv6 packets. IPv6 packets are tunneled in IPv4 with stateless v6 to v4 mapping and automatic prefix delegation derived from the v6 destination of each packet. The key component changes are to the routed CPE to make it 6rd capable, via software or hardware upgrade and introduction of a 6rd Border Relay function in the ISP network to route the packets to IPv6 networks.

Juniper Networks M Series Multiservice Edge Routers/T Series Core Routers/MX Series 3D Universal Edge Routers can play the role of BR.

4 Terminology
• 6rd prefix: This is an IPv6 prefix selected by the service provider for use by a 6rd domain. There is exactly one 6rd prefix for a given 6rd domain. An SP can deploy 6rd with a single 6rd domain or multiple 6rd domains.
• 6rd Customer Edge (6rd CE): This device functions as a CE router in a 6rd deployment, which is also called residential gateway (RG) or customer premises equipment (CPE).
• 6rd domain: A set of 6rd CEs and BRs is connected to the same virtual 6rd link. A service provider can deploy 6rd with a single domain or multiple 6rd domains based on geographic or operational boundaries. Each domain requires a separate 6rd prefix.
• CE LAN side: The functionality of a 6rd CE device serves the “customer-facing” side of the CE device. The CE LAN-side interface is fully IPv6 enabled.
• CE WAN side: The functionality of a 6rd CE device serves the “service provider-facing” side of the CE device. The CE WAN side is IPv4-only.
• 6rd Border Relay: A 6rd-enabled router is at the edge of a 6rd domain.
• BR IPv4 address: This is the IPv4 address of the 6rd Border Relay for a given 6rd domain. The CE device uses this IPv4 address to send packets to the BR in order to reach IPv6 destinations outside of the 6rd domain.
• CE IPv4 address: This is the IPv4 address given to the CE device as part of normal IPv4 Internet access (that is, configured via DHCP, PPP, or otherwise). This address can be global or private [RFC1918] within the 6rd domain.
• S-PIC/Service DPC: This refers to services PIC or Services DPC on the BR router where IPv6-in-IPv4 where the softwire is created.
5 Implementation: Juniper 6rd Solution

Juniper has implemented a 6rd Border Relay in its services DPC/PIC. IPv4 packets coming from the CE WAN side are steered to a Services DPC where a softwire is created. The IPv4 encapsulation is then removed to extract the inner IPv6 packets, which are then forwarded to IPv6 Internet. All these functions are performed in a single pass of the S-PIC/services DPC. Figure 1 illustrates the packet flow on the BR.

In the reverse path, IPv6 packets are sent to the Services DPC where they are encapsulated in an IPv4 packet corresponding to the proper softwire and sent to the 6rd CE.

The BR automatically creates softwires as the IPv4 packets are received from the CE router side, or gratuitous IPv6 packets are received from the Internet. A 6rd softwire on the Services DPC is identified by the 3-tuple containing service set ID, CE IPv4 address, and softwire concentrator IPv4 address. In addition, IPv6 flows are also created for the inner IPv6 payload, and they are associated with the specific softwire that carried them in the first place. When the last IPv6 flow associated with a softwire is destroyed, the softwire itself goes away. Therefore, the configuration is simplified, and there is no need to create or manage tunnel interfaces.

The softwire construct does not use any interface resources by design. 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 the platform (MS-PICMS-DPC) can support.

6 6rd Feature Summary

The following 6rd features are supported:

- 6rd domain routes and softwire concentrator routes can be exported via routing protocols.
- 6rd softwire concentrator IPv4 address can be pinged from the v4 access network or the CE device.
- Hairpinning traffic occurs between two 6rd domains.
- 6rd anycast is where a 6rd domain can be hosted on multiple S-PICs with one S-PIC actively handling all 6rd traffic at any given time. 6to4: 6to4 is an IPv4 tunnel-based transition mechanism defined in RFC-3056. It was designed to allow different IPv6 domains communicate with other IPv6 domains through IPv4 clouds without explicit IPv4 tunnels.
- All features are supported with both interface style and next-hop style service sets. For more information on interface-style and next-hop style service sets, please refer to Junos OS documentation.

In the following sections, these features are described with configuration examples and show CLI outputs.
7 Configuration Examples

In this section, an MX Series router is configured with an MS-DPC as a 6rd Border Relay. The 6rd feature is supported in a layer 3 package only. Assume that the PIC is in FPC 3 slots 0 and 1:

```
user@router# show chassis
fpc 3 {
    pic 0 {
        adaptive-services {
            service-package layer-3;
        }
    }
}
fpc 3 {
    pic 1 {
        adaptive-services {
            service-package layer-3;
        }
    }
}
```

This is common to all examples described in this document.

7.1 Topology

The following topology is used throughout this document when describing various scenarios.

![Topology Illustrating Multiple 6rd Domains](image-url)

In Figure 2, BR is the MX Series router with Services DPCs. The interface ge-1/2/0 is the ingress interface for 6rd domain 3040::0/16 (D1), and the interface ge-1/2/1 is the ingress interface for 6rd domain 3050::0/16 (D2) and 6rd domain 2002::0/16 (D3). The interface ge-1/2/2 is connected to the IPv6 Internet. For D1, 10.10.10.2 is the softwire initiator address and 30.30.30.1 is the softwire concentrator address; for D2, 40.40.40.2 is the softwire initiator address and 60.60.60.1 is the softwire concentrator address; and for D3, 160.160.160.2 is the softwire initiator address and 180.180.180.1 is the softwire concentrator address. The S-PICs used in all the examples are sp-3/0/0 and sp-3/1/0.
7.2 Basic 6rd Configuration

To configure 6rd, the user must configure the softwire rule. A sample softwire rule looks like the following:

```
[edit services]
user@router# show softwire
softwire-concentrator {
    v6rd v6rd-dom1 {
        softwire-address 30.30.30.1;
        ipv4-prefix 10.10.0.0/24;
        v6rd-prefix 3040::0/16;
        mtu-v4 1492;
    }
}
rule v6rd-dom1-r1 {
    match-direction input;
    term t1 {
        then {
            v6rd v6rd-dom1;
        }
    }
}
```

First, a softwire concentrator of type 6rd must be created with its associated IPv4 address. The softwire concentrator IPv4 address need not be the same as the ingress media interface. In fact, the softwire concentrator and the ingress media interface don’t need to be on the same network. The softwire concentrator is anchored on an IPv4 address independent of any interface and on a different prefix. The softwire concentrator object also has 6rd domain’s IPv6 prefix and IPv4 prefix of the CE WAN network. The mtu-v4 attribute sets the maximum transmission unit when encapsulating IPv6 packets into IPv4. In general, v6-in-v4 tunneling reduces the maximum transmission unit (MTU) of an IPv6 interface by 20 octets, that is if the basic IPv4 encap header does not contain optional fields. The administrator must configure the IPv6 interface MTU by taking the IPv4 MTU into account and subtracting at least 20 from that to allow room for the IPv4 header. As the services NPU doesn’t detect which egress interface the packet goes out on, the administrator must configure the IPv4 MTU under the softwire concentrator hierarchy by looking at the routing information. If, after adding the IPv4 header, the packet size grows beyond the IPv4 MTU configured, then the packet is dropped by the services NPU. This knob is mandatory since it depends on other network parameters under the administrator control.

Once a softwire concentrator is created, a softwire rule must be created to use it. The aforementioned rule basically states that any traffic destined to the softwire concentrator address of v6rd-dom1 results in the creation of a new softwire, if none exists already.

Once a softwire rule is created, it must be used in the service set as in the following:

```
service-set v6rd-dom1-service-set {
    ... 
    ... 
    softwire-rules v6rd-dom1-r1
    interface-service {
        service-interface sp-3/0/0;
    }
}
```
7.3 6rd Routes
Once 6rd is configured, the services PIC daemon (spd) on the routing engine installs routes for the softwire concentrator IPv4 address and 6rd domain's IPv6 prefix so that they point to the S-PIC. For the previous example, the following routes are installed:

```
user@router# run show route 30.30.30.1
inet.0: 49 destinations, 51 routes (48 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
30.30.30.1/32      *[Static/786432] 01:47:59
   Service to v6rd-dom1-service-set

[edit]
user@router# run show route 3040::0/16
inet6.0: 33 destinations, 42 routes (33 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
3040::/16          *[Static/786432] 01:48:14
   Service to v6rd-dom1-service-set
```

7.4 Advertising 6rd Routes
The 6rd routes installed by the services daemon (as shown in the previous section) can be advertised by configuring the proper routing policy. In this section, the relevant configuration is shown for both interface style and next-hop style service sets with VRFs.

7.4.1 Interface Style Service Set
First, the 6rd service set must be configured as shown in section 7.2.

```
[edit services]
user@router# show
   service-set v6rd-dom1-service-set {
      softwire-rules v6rd-dom1-r1;
      stateful-firewall-rules r1;
      interface-service {
         service-interface sp-3/0/0;
      }
   }
   stateful-firewall {
      rule r1 {
         match-direction input-output;
         term t1 {
            then {
               accept;
            }
         }
      }
   }
   softwire {
      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 rd1;
}
}
}

At this point, the service PIC daemon (spd) installs routes for 30.30.30.1 and 3040::0/16, pointing to the service PIC (sp-3/0/0 in this example). However, these routes are not advertised by default. You need to configure the routing policy to export these routes and apply the policy to the appropriate routing protocol. An example routing policy with OSPF is shown in the following.

```
[edit policy-options]
user@router# show
  policy-statement ospf-exp {
    term a {
      from protocol static;
      then accept;
    }
  }
```

Note that a more conservative routing policy can be configured with exact prefixes (in the from clause) that the user wants to export. In this case, all statically configured routes are exported, and since all services PIC daemon installed routes are static routes, this exports 6rd routes as well. Once the routing policy is in place, apply that to OSPF as in the following:

```
[edit protocols]
user@router#show
  ospf {
    export ospf-exp;
    area 0.0.0.0 {
      interface all;
      interface fxp0.0 {
        disable;
      }
    }
  }
  ospf3 {
    export ospf-exp;
    area 0.0.0.0 {
      interface all;
      interface fxp0.0 {
        disable;
      }
    }
  }
```

The OSPF hierarchy is needed to export v4 routes (in this case, 30.30.30.1), and the ospf3 hierarchy is needed to export v6 routes (in this case, 3040::/16).
### 7.4.2 Next-hop Style Service Set

```plaintext
[edit interfaces]
user@router#show
interfaces {
sp-3/0/0 {
    unit 0 {
        family inet;
        family inet6;
    }
    unit 1 {
        family inet;
        family inet6;
        service-domain inside;
    }
    unit 2 {
        family inet;
        family inet6;
        service-domain outside;
    }
}

[edit protocols]
user@router#show
ospf {
    export ospf-exp;
    area 0.0.0.0 {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
}

[edit policy-options]
user@router# show
policy-statement ospf-exp {
    term a {
        from protocol static;
        then accept;
    }
    term b {
        then reject;
    }
}

[edit routing-instances]
user@router# show
vrf1 {
    instance-type vrf;
    interface ge-1/2/0.0;
    interface sp-3/0/0.1;
    route-distinguisher 1.1.1.1:1;
    vrf-target target:100:100;
    routing-options {
        interface-routes {
            family inet6 {
                export {
```
point-to-point;
}
}
family inet {
export {
point-to-point;
}
}
}
}
router-id 5.5.5.5;
}
protocols {
ospf {
export ospf-exp;
area 0.0.0.0 {
interface all;
}
}
}
}
[edit services]
user@router# show
service-set ss1 {
  syslog {
    host local {
      services any;
    }
  } stateful-firewall-rules r1;
  softwire-rules v6rd-dom1-r1;
  next-hop-service {
    inside-service-interface sp-3/0/0.1;
    outside-service-interface sp-3/0/0.2;
  }
}
softwire {
  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;
      }
    }
  }
}
7.5 Pinging 6rd Softwire Concentrator

The softwire concentrator address itself can be pinged from a local routing engine. Or if these routes are advertised, this can be pinged from any host in the IPv4 access network (CE WAN side) or from the CE device itself.

```
user@router# run ping 30.30.30.1
PING 30.30.30.1 (30.30.30.1): 56 data bytes
64 bytes from 30.30.30.1: icmp_seq=0 ttl=255 time=0.368 ms
64 bytes from 30.30.30.1: icmp_seq=1 ttl=255 time=0.385 ms
^C
--- 30.30.30.1 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
```

7.6 6rd Domain to Internet

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

7.6.1 Interfaces Configuration

D1 Ingress interface:
```
ge-1/2/0 {
  unit 0 {
    family inet {
      service {
        input {
          service-set v6rd-dom1-service-set;
        }
        output {
          service-set v6rd-dom1-service-set;
        }
      }
      address 10.10.10.1/24;
    }
    family inet6;
  }
}
```

Egress Interface (IPv6 Internet):
```
ge-1/2/2 {
  unit 0 {
    family inet6 {
      address 3ABC::1/16;
    }
  }
}
```

Services PIC:
```
sp-3/0/0 {
  unit 0 {
    family inet;
    family inet6;
  }
}
```
Note that both inet and inet6 families are configured on the ingress media interface and S-PIC. Also, service is applied only on inet family on the ingress media interface.

### 7.6.2 Service Set Configuration

```yaml
service-set v6rd-dom1-service-set {
    softwire-rules v6rd-dom1-r1;
    stateful-firewall-rules r1;
    interface-service {
        service-interface sp-3/0/0;
    }
}
```

### 7.6.3 Softwire Concentrator Configuration

```yaml
[edit services]
user@router# show softwire
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 {
        v6rd v6rd-dom1;
    }
}
```

Here, softwire-address 30.30.30.1 is the softwire concentrator (BR) IPv4 address, 10.10.10.0/24 is the IPv4 prefix of the CE WAN side, and 3040::0/16 is the 6rd domain D1’s v6 prefix.

### 7.6.4 Stateful Firewall Configuration

```yaml
[edit services]
user@router# show stateful-firewall
rule r1 {
    match-direction input-output;
    term t1 {
        accept;
    }
}
```

It is mandatory to configure a stateful firewall rule that accepts all traffic in both input and output direction for 6rd to work. However, this is not enforced through the CLI. This is because in IPv6, gratuitous v6 packets are expected (due to anycast) to be detected and should not be dropped. So, S-PIC can detect reverse traffic without discovering forward traffic at all. This can also happen when S-PIC switchover occurs in the middle of a session. By default, the stateful firewall on the S-PIC drops all the traffic unless a rule is configured explicitly to allow it. Adding a stateful firewall rule prevents traffic drops in the event of a service PIC/MS-DPC switchover.

This example shows how to configure 6rd so that a client in the 6rd domain (D1) can communicate with a server in another 6rd domain (D2) (or vice versa).
7.6.5 Interfaces Configuration

D1 Ingress interface:
ge-1/2/0 {
    unit 0 {
        family inet {
            service {
                input {
                    service-set v6rd-dom1-service-set;
                }
                output {
                    service-set v6rd-dom1-service-set;
                }
            }
            address 10.10.10.1/24;
        }
        family inet6;
    }
}
D2 Ingress interface:
ge-1/2/1 {
    unit 0 {
        family inet {
            service {
                input {
                    service-set v6rd-dom2-service-set;
                }
                output {
                    service-set v6rd-dom2-service-set;
                }
            }
            address 40.40.40.1/24;
        }
        family inet6;
    }
}
Egress Interface (IPv6 Internet):
ge-1/2/2 {
    unit 0 {
        family inet6 {
            address 3ABC::1/16;
        }
    }
}
Services PIC:
sp-3/0/0 {
    unit 0 {
        family inet;
        family inet6;
    }
}
7.6.6 Service Set Configuration

```plaintext
service-set v6rd-dom1-service-set {
  softwire-rules v6rd-dom1-r1;
  stateful-firewall-rules r1;
  interface-service {
    service-interface sp-3/0/0;
  }
}

service-set v6rd-dom2-service-set {
  softwire-rules v6rd-dom2-r1;
  stateful-firewall-rules r1;
  interface-service {
    service-interface sp-3/0/0;
  }
}
```

7.6.7 Softwire Concentrator Configuration

```plaintext
[edit services]
user@router# show softwire
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;
  }
}
softwire-concentrator {
  v6rd v6rd-dom2 {
    softwire-address 60.60.60.1;
    ipv4-prefix 40.40.40.0/24;
    v6rd-prefix 3050::0/16;
    mtu-v4 9192;
  }
}
rule v6rd-dom1-r1 {
  match-direction input;
  term t1 {
    then {
      v6rd v6rd-dom1;
    }
  }
}
rule v6rd-dom2-r1 {
  match-direction input;
  term t1 {
    then {
      v6rd v6rd-dom2;
    }
  }
}
```
7.6.8 Stateful Firewall Configuration

[edit services]
user@router# show stateful-firewall
rule r1 {
  match-direction input-output;
  term t1 {
    then {
      accept;
    }
  }
}

7.7 6to4 Using 6rd

This example describes how to implement 6to4 using the 6rd feature. In this example, 6rd domain D3 has an IPv6 prefix of 2002::0/16, which is a well-known 6to4 prefix.

7.7.1 Interfaces Configuration

D1 Ingress interface:
ge-1/2/1 {
  unit 0 {
    family inet {
      service {
        input {
          service-set v6rd-dom3-service-set;
        }
        output {
          service-set v6rd-dom3-service-set;
        }
      }
      address 160.160.160.1/24;
    }
    family inet6;
  }
}

Egress Interface (IPv6 Internet):
ge-1/2/2 {
  unit 0 {
    family inet6 {
      address 3ABC::1/16;
    }
  }
}

Services PIC:
sp-3/0/0 {
  unit 0 {
    family inet;
    family inet6;
  }
}
7.7.2 Service Set Configuration

```
service-set v6rd-dom3-service-set {
  softwire-rules v6rd-dom3-r1;
  stateful-firewall-rules r1;
  interface-service {
    service-interface sp-3/0/0;
  }
}
```

7.7.3 Softwire Concentrator Configuration

```
[edit services]
user@router# show softwire
softwire-concentrator {
  v6rd v6rd-6to4-dom3 {
    softwire-address 180.180.180.1;
    ipv4-prefix 160.160.160.0/24;
    v6rd-prefix 2002::0/16;
    mtu-v4 9192;
  }
}
rule v6rd-dom3-r1 {
  match-direction input;
  term t1 {
    then {
      v6rd v6rd-6to4-dom3;
    }
  }
}
```

Note that you can have multiple terms in each softwire rule. For example, you can configure one softwire rule for both D2 and D3 like the following:

```
rule v6rd-dom2-dom3-r1 {
  match-direction input;
  term t1 {
    then {
      v6rd v6rd-dom2;
    }
  }
  term t2 {
    then {
      v6rd v6rd-6to4-dom3;
    }
  }
}
```

7.7.4 Stateful Firewall Configuration

```
[edit services]
user@router# show stateful-firewall
rule r1 {
  match-direction input-output;
  term t1 {
    then {
      accept;
    }
  }
}
```
7.8 6rd Anycast (Redundancy)

The 6rd anycast feature is used to configure multiple S-PICs to back up a 6rd domain’s traffic. However, for a given 6rd domain, only one S-PIC is actively processing the traffic at any point in time (called primary PIC). Other PICs are in standby mode, but these standby S-PICs can actively process 6rd traffic of some other 6rd domains (possibly provide other services such as NAT, etc.). When the primary S-PIC goes down for some reason, traffic is immediately routed to one of the standby S-PICs. When 6rd anycast is configured, multiple routes are installed for the 6rd softwire concentrator address and v6rd domain prefix, each pointing to different S-PIC. The following subsections show configuration details where 6rd domains D1 and D3 are hosted on two S-PICs, sp-3/0/0 and sp-3/1/0.

### 7.8.1 Interfaces Configuration

<table>
<thead>
<tr>
<th>D1 and D3 Ingress interface: ge-1/2/0 {</th>
</tr>
</thead>
<tbody>
<tr>
<td>unit 0 {</td>
</tr>
<tr>
<td>family inet {</td>
</tr>
<tr>
<td>service {</td>
</tr>
<tr>
<td>input {</td>
</tr>
<tr>
<td>service-set v6rd-dom1-dom3-service-set;</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>output {</td>
</tr>
<tr>
<td>service-set v6rd-dom1-dom3-service-set;</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>address 10.10.10.1/24;</td>
</tr>
<tr>
<td>address 160.160.160.1/24;</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>family inet6;</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>Egress Interface (IPv6 Internet): ge-1/2/2 {</td>
</tr>
<tr>
<td>unit 0 {</td>
</tr>
<tr>
<td>family inet6 {</td>
</tr>
<tr>
<td>address 3ABC::1/16;</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>Services PIC: sp-3/0/0 {</td>
</tr>
<tr>
<td>unit 0 {</td>
</tr>
<tr>
<td>family inet;</td>
</tr>
<tr>
<td>family inet6;</td>
</tr>
<tr>
<td>}</td>
</tr>
</tbody>
</table>
7.8.2 Service Set Configuration

```plaintext
service-set v6rd-dom1-dom3-service-set {
    syslog {
        host local {
            services any;
        }
    }
    softwire-rules v6rd-dom1-dom3-r1;
    stateful-firewall-rules r1;
    interface-service {
        service-interface sp-3/0/0;
    }
}

service-set v6rd-anycast-dom1-dom3-service-set {
    syslog {
        host local {
            services any;
        }
    }
    softwire-rules v6rd-dom1-dom3-r1;
    stateful-firewall-rules r1;
    interface-service {
        service-interface sp-3/1/0;
    }
}
```

Note that both the previous service sets use the softwire rule, which refers to the same 6rd domains.

7.8.3 Softwire Concentrator Configuration

```plaintext
[edit services]
user@router# show softwire
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;
}

v6rd v6rd-6to4-dom3 {
    softwire-address 180.180.180.1;
    ipv4-prefix 160.160.160.0/24;
    v6rd-prefix 2002::0/16;
    mtu-v4 9192;
}

rule v6rd-dom1-dom3-r1 {
    match-direction input;
    term t1 {
        then {
            v6rd v6rd-dom1;
        }
    }
    term t2 {
        then {
            v6rd v6rd-6to4-dom3;
        }
    }
}
```

Note that both D1 and D3 are configured in the same rule, but they are used in two different service sets. Because of this configuration, 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-dom1-dom3-service-set
   [Static/851968] 00:24:11
   Service to v6rd-anycast-dom1-dom3-service-set

[edit]
user@router# run show route 180.180.180.1
inet.0: 43 destinations, 46 routes (42 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

180.180.180.1/32  *[Static/786432] 00:24:21
   Service to v6rd-dom1-dom3-service-set
   [Static/851968] 00:24:21
   Service to v6rd-anycast-dom1-dom3-service-set

[edit]
user@router# run show route 2002::0/16
inet6.0: 23 destinations, 33 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

2002::/16  *[Static/786432] 00:24:27
   Service to v6rd-dom1-dom3-service-set
   [Static/851968] 00:24:27
   Service to v6rd-anycast-dom1-dom3-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-dom1-dom3-service-set
   [Static/851968] 00:24:39
   Service to v6rd-anycast-dom1-dom3-service-set
```

In the forward direction (from D1/D3 to IPv6 Internet), the traffic hits the service filter (that is, the service set is applied to the ingress interface) and comes to S-PIC sp-3/0/0. In the reverse direction (from IPv6 Internet to D1/D3), the traffic hits the route for 3040::0/16 or 2002::0/16 and comes to the service S-PIC sp-3/0/0 because the metric value for those routes is numerically smaller, hence and preferred, compared to the routes pointing to sp-3/1/0 for the same. When sp-3/0/0 goes down, the routes pointing to sp-3/0/0 are deleted and therefore all traffic is now sent to sp-3/1/0. As there is no state synchronization between S-PICs and 6rd is supposed to be stateless, packet loss is minimal.

Note that if you apply `anycast-v6rd-dom1-dom3-service-set` instead of `v6rd-dom1-dom3-service-set` to the ingress interface, the forward traffic goes to sp-3/1/0 because of the service filter and the reverse traffic still goes to sp-3/0/0. Note that you can also apply both service sets to the ingress interface, in which case the service filter that the forward traffic hits depends on the order in which the service sets are applied to the ingress interface.
7.8.4 Stateful Firewall Configuration

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

7.9 6rd + DS-Lite in Same Service Set

Juniper offers multiple transition mechanisms to address IPv4 address exhaustion. The extensive feature toolkit includes support for NAT444, DS-Lite, and 6rd among other features. During transition to IPv6, or in an effort to enable IPv6 services over IPv4 infrastructure, there might be situations where a provider might need to deploy both DS-Lite and 6rd. 6rd can be configured with DS-Lite on the same PIC and in the same service set. The following subsections show configuration details of such a scenario where both DS-Lite and 6rd are configured in the same service set.

7.9.1 Interfaces Configuration

Ingress interface:

```
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) family of subunit 0 as well as the inet6 (IPv6) family. This way, both DS-Lite IPv6 traffic and 6rd IPv4 traffic hit the service filter and are sent to the services PIC.

**Egress Interface (IPv6 Internet):**

```
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.

**Services PIC:**

```
sp-3/0/0 {
  unit 0 {
    family inet;
    family inet6;
  }
}
```

**7.9.2 Service Set Configuration**

```
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. In addition, there is a softwire rule for DS-Lite and a NAT rule to translate 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.
7.9.3 Softwire Concentrator Configuration

```
[edit services]
user@router# show softwire
softwire-concentrator {
  ds-lite dsl1 {
    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;
  }
}
rule v6rd-r1 {
  match-direction input;
  term t1 {
    then {
      v6rd v6rd-dom1;
    }
  }
}
rule dslite-r1 {
  match-direction input;
  term dslite-t1 {
    then {
      ds-lite dsl1;
    }
  }
}
```

Because of this configuration, 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

[edit]
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
```
7.9.4 Stateful Firewall Configuration

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

7.9.5 NAT Configuration

```
[edit services]
user@router# show 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.

```
[edit]
user@router# run 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
```
8 Configuration Tips

Currently, there is no support for fragmentation and reassembly. Therefore, the administrator needs to properly configure the MTUs on the IPv6 and IPv4 network.

9 Operations and Management

There are two classes of commands that can be used to check the workings of the 6rd implementation: CLI and debug. The debug commands are used from the services DPC/PIC console, and the CLI follows the traditional Juniper implementation.

9.1 6rd CLI

The administrator has several commands available to perform operations and management. The following sections discuss a few of them. The user is referred to the appropriate Junos OS manual for a more complete set.

9.1.1 Flows and Conversations

The administrator can use the command “show services stateful-firewall flows” to check the creation of the softwires and flows. An example output for the configuration previously discussed is shown in the following.

```
user@router# run show services stateful-firewall flows
Interface: sp-3/0/0, Service set: v6rd-dom1-dom3-service-set
Flow     State     Dir     Frm count
TCP      3abc::2:80  ->  3040:200::2:1025 Forward O       52017
Softwire 10.10.10.2  ->  30.30.30.1
TCP      3040:200::2:1025  ->  3abc::2:80 Forward I       57071
Softwire 10.10.10.2  ->  30.30.30.1
IPV6     10.10.10.2  ->  30.30.30.1:0 Forward I          0
Softwire 10.10.10.2  ->  30.30.30.1
```

Another important command shows the conversations (collection of related flows).

```
user@router# run show services stateful-firewall conversations
Interface: sp-3/0/0, Service set: v6rd-dom1-dom3-service-set
Conversation: ALG protocol: tcp
Number of initiators: 1, Number of responders: 1
Flow     State     Dir     Frm count
TCP      3040:200::2:1026  ->  3abc::2:80 Forward I       717077
Softwire 10.10.10.2  ->  30.30.30.1
TCP      3abc::2:80  ->  3040:200::2:1026 Forward O       717171
Softwire 10.10.10.2  ->  30.30.30.1
```
9.2 Softwire

user@router# run show services softwire statistics
sp-3/0/0
DS-Lite Statistics:

  Softwires Created: 0
  Softwires Deleted: 0
  Flows Created: 0
  Flows Deleted: 0
  Slow Path: 0
  Fast Path: 0
  Fast Path Encapsulated: 0
  Rule Match Failed: 0
  Rule Match Succeeded: 0
  IPv6 Packets Fragmented: 0
  Softwire Creation Failed: 0
  Flow Creation Failed: 0
  Flow Creation Failed - Retry: 0
  Slow Path Failed: 0
  Slow Path Failed - Retry: 0
  Packet not IPv4 in IPv6: 280584157
  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

6rd Statistics:

  Softwires Created: 36
  Softwires Deleted: 35
  Flows Created: 30
  Flows Deleted: 35
  Slow Path: 33
  Fast Path: 294924475
  Fast Path Encapsulated: 280584155
  Rule Match Failed: 0
  Rule Match Succeeded: 13390
  Softwire Creation Failed: 0
  Flow Creation Failed: 0
  Flow Creation Failed - Retry: 6
  Slow Path Failed: 0
  Slow Path Failed - Retry: 6
  Packet not IPv6 in IPv4: 0
  Slow Path Failed - IPv6 Next Header Offset: 35
  Decapsulated Packet not IPv6: 0
  Encapsulation Failed - No space for Outer Header: 0
  No Softwire ID: 0
  No Flow Extension: 0

user@router# run show services softwire
Interface: sp-3/0/0, Service set: v6rd-dom1-dom3-service-set
Software | Direction | Flow count
----------|-----------|----------
10.10.10.2 | -> | 30.30.30.1 | I | 13
9.3 Debug CLI

Please refer to the appendix for CLI commands related to 6rd operations.

10 ALG Support

Currently, 6rd supports ICMPv6 ALG. The following CLI output shows flows created when a host in the 6rd domain D1 pings a server in the IPv6 Internet.

```
user@router# run show services stateful-firewall flows
Interface: sp-3/0/0, Service set: v6rd-dom1-dom3-service-set
Flow                              State   Dir       Frm count
ICMPV6 3040:200::2        ->        3abc::1       Watch    I               9
Softwire 10.10.10.2         ->      30.30.30.1
IPV6    10.10.10.2:0     ->     30.30.30.1:0     Forward  I               0
Softwire 10.10.10.2         ->      30.30.30.1
ICMPV6 3abc::1        ->    3040:200::2       Watch    O              10
Softwire 10.10.10.2         ->      30.30.30.1
```

11 Summary

This implementation guide provides an overview of the 6rd feature set and its application. The guide also provides basic configuration guidelines to deploy 6rd with services PIC or MS-DPC blade, and it addresses some operational aspects of deployment. It also shows an example of coexistence of DS-Lite and 6rd features. For more information, please visit www.juniper.net/ipv6.

12 Appendix

Following is list of CLI commands that are helpful with operations when configuring and deploying 6rd. The topology and complete router configs are at the end of this appendix (see Figure 3).

1) show services softwire
2) show services softwire count
3) show services softwire flows count
4) show services softwire statistics
5) show services softwire statistics v6rd
6) show services softwire flows
7) show services softwire flows v6rd concentrator <concentrator ip>
8) show services softwire flows v6rd initiator<initiator ip>
9) show services softwire flows v6rd initiator <initiator ip> concentrator<concentrator ip>
10) show services softwire flows v6rd concentrator<concentrator ip> initiator <initiator ip>

Commands Detail:

1) CLI Syntax: show services softwire

Description: This CLI will display all the softwires created for a service set and number of flow count in that softwire.

Example:

```
regress@sledding# run show services softwire
Interface: sp-0/0/0, Service set: dslite-svc-set1
Softwire                  Direction Flow count
2001::2        ->        1001::1 I           3
2001::3        ->        1001::1 I           3
```
2) CLI Syntax: show services softwire count
Description: This CLI will display all the current softwire count for a service set for both Ds-Lite and V6rd.

Example:

```
regress@sledding# run show services softwire count
```

<table>
<thead>
<tr>
<th>Interface</th>
<th>Service set</th>
<th>DS-Lite</th>
<th>6RD</th>
</tr>
</thead>
<tbody>
<tr>
<td>sp-0/0/0</td>
<td>dslite-svc-set1</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

3) CLI Syntax: show services softwire flows count
Description: This CLI will display flow count for per service set.

Example:

```
regress@sledding# run show services softwire flows count
```

<table>
<thead>
<tr>
<th>Interface</th>
<th>Service set</th>
<th>Flow count</th>
</tr>
</thead>
<tbody>
<tr>
<td>sp-0/0/0</td>
<td>dslite-svc-set1</td>
<td>6</td>
</tr>
</tbody>
</table>

4) CLI Syntax: show services softwire statistics
Description: This CLI will display statistics for both Ds-Lite and V6rd for each service pic.

Example:

```
regress@sledding# run show services softwire statistics
```

**DS-Lite Statistics:**

```
Service PIC Name: :sp-0/0/0

Statistics
----------
Softwires Created: 2
Softwires Deleted: 0
Softwires Flows Created: 2
Softwires Flows Deleted: 0
Slow Path Packets Processed: 2
Fast Path Packets Processed: 1043786
Fast Path Packets Encapsulated: 1450803
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
5) CLI Syntax: show services softwire statistics v6rd

Description: This CLI will display statistics of v6rd for each service pic.

Example:

```bash
regress@billiard> show services softwire statistics v6rd
6rd Statistics:
Service PIC Name                     : sp-0/1/0
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

Service PIC Name                        : sp-1/3/0
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

7) CLI Syntax: show services softwire flows

Description: This CLI will display all the flows which uses softwire in it.

Example:

regress@sledding# run show services softwire flows
Interface: sp-0/0/0, Service set: dslite-svc-set1
Flow count
TCP 200.200.200.2:80 -> 33.33.33.1:1066 Forward O 2005418
NAT dest 33.33.33.1:1066 -> 20.20.1.2:1025
Softwire 1001::1 -> 2001::2
TCP 20.20.1.2:1025 -> 200.200.200.2:80 Forward I 2007168
NAT source 20.20.1.2:1025 -> 33.33.33.1:1066
Softwire 2001::2 -> 1001::1
TCP 20.20.1.2:1025 -> 200.200.200.2:80 Forward I 2635998
NAT source 20.20.1.2:1025 -> 33.33.33.1:1065
Softwire 2001::3 -> 1001::1
DS-LITE 2001::2 -> 1001::1 Forward I 2008157
TCP 200.200.200.2:80 -> 33.33.33.1:1065 Forward O 2637909
NAT dest 33.33.33.1:1065 -> 20.20.1.2:1025
Softwire 1001::1 -> 2001::3
DS-LITE 2001::3 -> 1001::1 Forward I 2640499

8) CLI Syntax: show services softwire flows v6rd concentrator <concentrator ip>

Description: This CLI will filter out flows from CLI "show softwire services flows" based on specified v6rd concentrator ip.

Example:

regress@ billiard # run show services softwire flows v6rd concentrator 2.2.2.2

9) CLI Syntax: show services softwire flows v6rd initiator<initiator ip>

Description: This CLI will filter out flows from CLI "show softwire services flows" based on specified v6rd initiator ip.

Example:

regress@ billiard # run show services softwire flows v6rd initiator 3.3.3.3
10) CLI Syntax: `show services softwire flows v6rd initiator<initiator ip>concentrator<concentrator ip>`

Description: This CLI will filter out flows from CLI “show softwire services flows” based on both specified v6rd initiator ip and concentrator ip.

Example:

```
regress@ billiard # run show services softwire flows v6rd initiator 3.3.3.3 concentrator 2.2.2.2
```

11) CLI Syntax: `show services softwire flows v6rd concentrator<concentrator ip>initiator<initiator ip>`

Description: This CLI will filter out flows from CLI “show softwire services flows” based on both specified v6rd initiator ip and concentrator ip.

Example:

```
regress@ billiard # run show services softwire flows v6rd concentrator 2.2.2.2 initiator 3.3.3.3
```

TOPOLOGY :

```
  +----------+           +----------+
  | (Client) |           | (Server) |
  Shenick ==--------------= M/Mx/T =--------------= Shenick
  |           |           | (AFTR) |
  IPv4 Stateful traffic | / |
  is encapsulated in a | (BR) |
  IPv6 tunnel | |
  (Acts as host + SI/B4) | |
  +----------+
```

![Figure 3: Topology used to illustrate CLI commands for 6rd deployment and operations](image)

Router CONFIG :

```
interfaces {
  sp-0/0/0 {
    services-options {
      inactive: syslog {
        host local {
          services any;
        }
      }
    }
  }
  unit 0 {
```

---

[Image](image) - TOPOLOGY used to illustrate CLI commands for 6rd deployment and operations
family inet;
family inet6;
}
unit 1 {
    family inet;
    family inet6;
    service-domain inside;
}
unit 2 {
    family inet;
    family inet6;
    service-domain outside;
}
}
ge-3/1/0 {
    unit 0 {
        family inet {
            address 200.200.200.1/24;
        }
    }
}
ge-3/1/5 {
    unit 0 {
        family inet;
        family inet6 {
            address 2001::1/16;
        }
    }
}
}
}
routing-options {
    rib inet6.0 {
        static {
            route 1001::1/128 next-hop sp-0/0/0.1;
        }
    }
}
}
}
}
services {
    service-set dslite-svc-set1 {
        syslog {
            host local {
                services any;
            }
        }
        softwire-rules dslite-rule;
        stateful-firewall-rules sfw-r1;
        nat-rules dslite-nat-rule1;
        next-hop-service {
            inside-service-interface sp-0/0/0.1;
            outside-service-interface sp-0/0/0.2;
        }
    }
    stateful-firewall {
        rule sfw-r1 {
            match-direction input;
            term t1 {
                from {
                    applications [ junos-http junos-ftp junos-rtsp junos-icmp-all
                }
            }
        }
    }
};
  
  then {
    accept;
    syslog;
  }
}
}
}
softwire {
  ipv6-multicast-interfaces all;
  softwire-concentrator {
    ds-lite ds1 {
      softwire-address 1001::1;
      mtu-v6 9192;
    }
  }
  rule dslite-rule {
    match-direction input;
    term t1 {
      then {
        ds-lite ds1;
      }
    }
  }
}

nat {
  pool dslite-pool1 {
    address-range low 33.33.33.1 high 33.33.33.32;
    port {
      automatic;
    }
  }
  rule dslite-nat-rule1 {
    match-direction input;
    term t1 {
      from {
        source-address {
          20.20.0.0/16;
        }
      }
      then {
        translated {
          source-pool dslite-pool1;
          translation-type {
            source dynamic;
          }
        }
      }
    }
  }
}

adaptive-services-pics {
  traceoptions {
    flag all;
  }
}
**Shenick Host Config:**

<table>
<thead>
<tr>
<th>Host</th>
<th>IP Address</th>
<th>Subnet Mask</th>
<th>Gateway</th>
<th>Virtual Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>server_gw4</td>
<td>Static 200.200.200.1</td>
<td></td>
<td>server_gw4</td>
<td></td>
</tr>
<tr>
<td>client_gw6</td>
<td>Static 2001::1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B4_2</td>
<td>Card 3/0</td>
<td>Static 2001::3/16</td>
<td>client_gw6</td>
<td>2001::1</td>
</tr>
<tr>
<td>B4_1</td>
<td>Card 3/0</td>
<td>Static 2001::2/16</td>
<td>client_gw6</td>
<td>2001::1</td>
</tr>
<tr>
<td>server</td>
<td>Card 3/1</td>
<td>Static 200.200.200.2/24</td>
<td>server_gw4</td>
<td>200.200.200.1</td>
</tr>
</tbody>
</table>

**Shenick App Config:**

<table>
<thead>
<tr>
<th>Server Type</th>
<th>Host</th>
<th>IP Address</th>
<th>Port</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP Server</td>
<td>server4</td>
<td>200.200.200.2/24</td>
<td>80</td>
<td>1 Application</td>
</tr>
<tr>
<td>HTTP Client1</td>
<td>client1_B4_1</td>
<td>20.20.1.2/16</td>
<td></td>
<td>dslite_server4</td>
</tr>
<tr>
<td>HTTP Client2</td>
<td>client2_B4_2</td>
<td>20.20.1.2/16</td>
<td></td>
<td>dslite_server4</td>
</tr>
</tbody>
</table>

---

13 Bibliography


About Juniper Networks

Juniper Networks is in the business of network innovation. From devices to data centers, from consumers to cloud providers, Juniper Networks delivers the software, silicon and systems that transform the experience and economics of networking. The company serves customers and partners worldwide. Additional information can be found at [www.juniper.net](http://www.juniper.net).