JUNIPER NETWORKS

# Network Configuration Example

## Virtual Router Use Case for Educational Networks

Release
13.1

Published: 2013-02-08

Copyright © 2013, Juniper Networks, Inc.

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

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

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

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

*Network Configuration Example  Virtual Router Use Case for Educational Networks*
Release 13.1
NCE0039
Copyright © 2013, Juniper Networks, Inc.
All rights reserved.

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

YEAR 2000 NOTICE

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

## END USER LICENSE AGREEMENT

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

# Table of Contents



Copyright © 2013, Juniper Networks, Inc.

Copyright © 2013, Juniper Networks, Inc.

## Introduction

This document provides an overview of virtual routers as implemented by a network user (Educational Network, in this document called “New University”) connected to multiple other networks to exchange traffic. This document also provides examples of configuring virtual router routing instances for network peering using Juniper Networks SRX Series Services Gateway and J Series Services Router.

## Using Virtual Routers to Provide Customer Peering

This topic includes the following sections:



### Virtual Router Overview

Multiple distinct routers that are supported within a single router allow service providers to configure multiple, separate, secure routers within a single chassis. These routers are called virtual routers. Applications for this feature include the creation of individual routers dedicated to wholesale customers, corporate virtual private network (VPN) users, or specific traffic type users.

Internet Service Providers (ISPs) can have vast numbers of customers, including organizations that have their own large networks (with users connected to each other on a local access network (LAN) through Ethernet or a similar connection). An Educational Network, called “New University” in the “Example: Configuring Virtual Routers for Educational Networks” on page 12 example, is one such organization that is connected to multiple other networks to exchange traffic. New University requires peering with multiple networks to exchange data between the users of each network for mutual benefit.

This example shows how to perform network peering between New University and multiple other networks.

The network topology used in this example consists of the traditional Internet, Internet2, and networks such as national and regional lambda rails. These networks are administratively separate and are not interconnected with each other.

The following terms are used in this example:

- Service provider networks – Transit-based networks that offer connectivity to other autonomous systems. For example: traditional Internet peering, Internet2 peering, and high-speed regional networks such as National Lambda Rails (NLRs).
- Educational Network – “New University” is the example used in this document.
- Customer networks – Networks connected to New University and that view New University as a network service provider of local and distant network connectivity.

Copyright © 2013, Juniper Networks, Inc.

## Virtual Router Topology for Educational Networks

Consider the network topology as shown in Figure 1 on page 2. In this example, three virtual router routing instances are created to accommodate the Internet, Internet2, and NLR peering sessions and prefixes. The service provider routers **Provider 1** and **Provider 2** are configured for virtual router **VR-Internet**. Similarly, two Internet2 (i2) networks are configured for virtual router **VR-i2**. The virtual router **VR-nlr** is connected to the NLR network.

### Figure 1: Sample of the Virtual Router Topology for New University

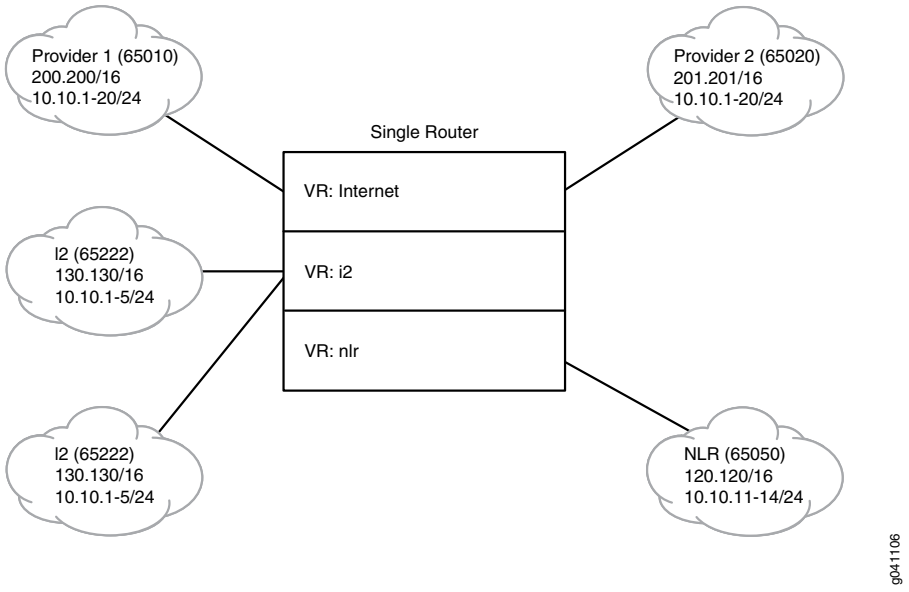



Table 1 on page 2 shows the mapping between devices, IP prefixes, and three network types available from different network providers.

### Table 1: Details of the Virtual Router Topology for New University

| Network Types | Devices | IP Prefixes Representing the Internet Routing Table |
|---|---|---|
| Internet | AS65010 and AS65020 | 10.10.1-20/24<br>200.200/16<br>201.201/16 |
| Internet2 | AS65222 | 130.130/16 |
| NLR | AS65050 | 120.120/16<br>0.10.11-14/24 |


Copyright © 2013, Juniper Networks, Inc.

Table 2 on page 3 provides complete mapping details between network types, virtual routers, customer routers, IP prefixes, and routing table details.

**Table 2: Provider Prefixes in the Virtual Router Topology**

| Network Types | Virtual Router Routing Instance | Routing Table | Peer Networks | IP Prefixes |
|---|---|---|---|---|
| Internet | VR-Internet | internet.inet.0 | internet-peer-1 | 10.10.1-20/24<br>200.200/16 |
| | | internet.inet.0 | internet-peer-2 | 10.10.1-20/24<br>201.201/16 |
| Internet2 | VR-i2 | i2.inet.0 | inet-peer-1 | 10.10.1-5/24<br>130.130/16 |
| | | i2.inet.0 | inet-peer-2 | 10.10.1-5/24<br>130.130/16 |
| NLR | VR-nlr | nlr.inet.0 | nlr-peer-1 | 10.10.11 14/24<br>120.120/16 |

## Routing Tables

As per the topology shown in Figure 1 on page 2, three routing tables are labeled using the convention <routing-instance-name> .inet.0.

The three routing tables used in this example are:

- internet.inet.0
- i2.inet.0
- nlr.inet.0

Because each network type has a distinct and separate routing table, routing information should be reviewed in each virtual router.

Table 3 on page 3 provides IP prefixes from provider networks available in each routing table.

**Table 3: Mapping of Routing Tables and Provider Prefixes**

| Routing Table | Peer | IP Prefixes |
|---|---|---|
| internet.inet.0 | internet-peer-1 | 10.10.1 20/24<br>200.200/16 |

Copyright © 2013, Juniper Networks, Inc.

Table 3: Mapping of Routing Tables and Provider Prefixes *(continued)*

| Routing Table | Peer | IP Prefixes |
|---|---|---|
| internet.inet.0 | internet-peer-2 | 10.10.1 20/24<br>201.201/16 |
| i2.inet.0 | inet-peer-1 | 10.10.1 5/24<br>130.130/16 |
| i2.inet.0 | inet-peer-2 | 10.10.1 5/24<br>130.130/16 |
| nlr.inet.0 | nlr-peer-1 | 10.10.11-14/24<br>120.120/16 |

Following is an example of the IP prefixes received from the peer (internet-peer-1):

```
user@J2350-2-R2# run show route receive-protocol bgp 172.16.0.6

internet.inet.0: 40 destinations, 63 routes (40 active, 0 holddown, 1 hidden)
Prefix Nexthop MED Lclpref AS path
* 10.10.1.0/24 172.16.0.6 65010 I
* 10.10.2.0/24 172.16.0.6 65010 I
* 10.10.3.0/24 172.16.0.6 65010 I
* 10.10.4.0/24 172.16.0.6 65010 I
* 10.10.5.0/24 172.16.0.6 65010 I
* 10.10.6.0/24 172.16.0.6 65010 I
* 10.10.7.0/24 172.16.0.6 65010 I
* 10.10.8.0/24 172.16.0.6 65010 I
* 10.10.9.0/24 172.16.0.6 65010 I
* 10.10.10.0/24 172.16.0.6 65010 I
* 10.10.11.0/24 172.16.0.6 65010 I
* 10.10.12.0/24 172.16.0.6 65010 I
* 10.10.13.0/24 172.16.0.6 65010 I
* 10.10.14.0/24 172.16.0.6 65010 I
* 10.10.15.0/24 172.16.0.6 65010 I
* 10.10.16.0/24 172.16.0.6 65010 I
* 10.10.17.0/24 172.16.0.6 65010 I
* 10.10.18.0/24 172.16.0.6 65010 I
* 10.10.19.0/24 172.16.0.6 65010 I
* 10.10.20.0/24 172.16.0.6 65010 I
* 200.200.0.0/16 172.16.0.6 65010 I
* 201.201.0.0/16 172.16.0.6 65010 65020 I
```

## Virtual Router Configuration Requirements

The basic steps to create a virtual router include:

- Creating virtual router routing instances
- Configuring protocols for provider and logical interfaces between participating routers
- Assigning interfaces to security zones


Copyright © 2013, Juniper Networks, Inc.

- Importing routes between virtual router routing instances
- Defining and applying security policies to virtual router routing instances

NOTE:

To create a virtual router you must:

- Configure separate logical interfaces between each of the service provider devices participating in a virtual router routing instance.
- Configure separate logical interfaces between the service provider devices and the customer devices participating in each routing instance.
- Configure a unique set of logical interfaces from all the participating routers.

This topic provides details on some aspects of configuring the virtual router routing instances and includes the following sections:



## BGP Communities

BGP is a routing protocol for communication between autonomous systems (ASs) on the Internet. BGP differentiates the prefixes from each service provider to use them in the routing policy.

Autonomous systems share a common set of prefixes and have a peer relationship together in the cloud. Each network type and associated peering sessions are placed into a virtual router that is representative of the network type.

A BGP community consists of routes that share a common attribute. A BGP community specifies a destination grouped into a logical unit to make it easier to apply routing policies. By performing global configurations for a community of routes, you reduce the tasks, time, CPU workload, and memory needed to set parameters for each route. Once you have established which routes you want to include in your community, you create a community list to which you add the IP addresses of the desired routes.

Table 4 on page 5 shows the BGP communities used in this example.

### Table 4: The BGP Community List

| Community Name | BGP Community |
|---|---|
| internet-vr | 65000:111 |

Copyright © 2013, Juniper Networks, Inc.

Table 4: The BGP Community List *(continued)*

| Community Name | BGP Community |
|---|---|
| i2-vr | 65000:222 |
| nlr-vr | 65000:333 |
| cust-routes | 65000:1000 |
| prepend-twice | 65000:500 |

The customer scenario examples in this document provide details on:

- Communities used by the routing policy
- Sharing of prefixes
- Forwarding of traffic in the network

Use the **show policy-options | match community | match member** command to display the routes that are permitted by the BGP community, as shown in the following sample output:

```
user@J2350-2-R2> show policy-options | match community | match member
community cust-routes members 65000:1000;
community i2-vr members 65000:222;
community internet-vr members 65000:111;
community nlr-vr members 65000:333;
community prepend-twice members 65000:500;
```

The AS-path statement is used with the routing policy to prevent New University from becoming a transit provider to the ISP networks. This statement is used in the policy to match the AS-path attribute in BGP advertisements. Because there are multiple peers per network type, it is important to prevent the network from advertising provider prefixes learned from one provider to other network providers.

Use the **null "()"** regular expression, which represents an empty value. When used in a BGP policy, the null value only allows prefixes originating from New University to be advertised to external BGP peers.

Use the **show policy-options | match as-path** command to verify the as-path value, as shown in the following sample output:

```
user@J2350-2-R2> show policy-options | match as-path
as-path null "()";
```

### RIB Groups

The routing information base (RIB) is a logical data structure used by BGP to store routing information that includes:

- Routes that BGP learned from peers
- Local routes resulting from the application of BGP policies to the learned routes


Copyright © 2013, Juniper Networks, Inc.

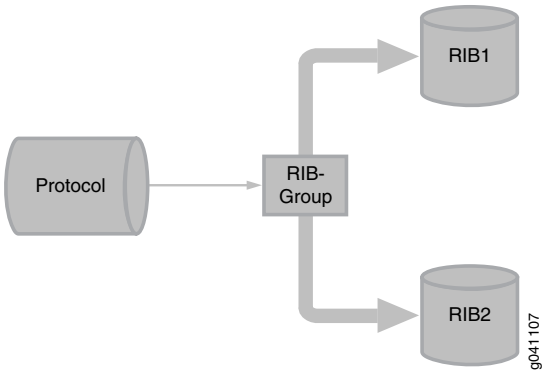
- Routes that BGP advertises to its peers

Sometimes routing information is stored in multiple RIBs. For example, New University to customer network interfaces are stored in multiple routing tables for next-hop resolution.

A RIB group includes one or more routing tables to form a routing table group. A routing protocol can import routes into all the routing tables in the group and can export routes from a single routing table. A RIB group determines how routes are distributed between RIBs.

Figure 2 on page 7 shows the relationship between the protocol, RIB groups, and routing tables.

### Figure 2: Relationship Between the Protocol, RIB Group, and Routing Tables




A RIB group only affects the protocols that it is configured under. The default behavior for a protocol such as OSPF is to place the routes into the main routing table (RIB). There are cases when the desired behavior is to distribute those OSPF routes between RIBs.

When a RIB group is applied to a protocol, such as OSPF, the RIB group distributes the OSPF routes to all of the RIBs specified in the RIB group. In this case, the learned OSPF routes are placed in the RIB-1 and RIB-2 routing tables.

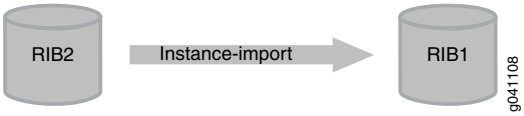
## Instance Import

The instance import statement is similar to the RIB group, but it affects the entire RIB for a virtual router routing instance instead of individual protocols.

Using the instance import statement essentially replicates the routes from one RIB to another. The replication can be constrained by policy.

Figure 3 on page 7 shows the role of the instance import statement in the RIB group.

### Figure 3: Instance Import




Copyright © 2013, Juniper Networks, Inc.

### Virtual Router for Internet Access (VR-Internet)

The virtual router for Internet access is a holding container for the peering sessions and prefixes that are representative of the traditional Internet.

Use the **show routing-instances *virtual-router-name*** command to display the configuration of the virtual router routing instance. The following sample output shows the configuration of the **internet** virtual router routing instance:

```
user@J2350-2-R2> show routing-instances internet

    instance-type virtual-router;
    interface ge-0/0/0.230;
    interface ge-0/0/0.270;
    interface lo0.1;
    protocols {
        bgp {
            group peer-1 {
                type external;
                advertise-inactive;
                import internet-import-policy;
                export internet-export-policy;
                peer-as 65010;
                multipath multiple-as;
                neighbor 172.16.0.6;
            }
            group peer-2 {
                type external;
                advertise-inactive;
                import internet-import-policy;
                export internet-export-policy;
                peer-as 65020;
                multipath multiple-as;
                neighbor 172.16.0.26;
            }
        }
    }
```

Use the **show policy-options policy-statement *policy-statement-name*** command to display the details of the import policy. The following sample output shows the configuration of the **internet-import-policy** virtual router routing instance. All prefixes learned through the external BGP (EBGP) neighbors are tagged with the community Internet virtual router (65000:111) by the BGP import policy **internet-import-policy**.

```
user@J2350-2-R2> show policy-options policy-statement internet-import-policy

    term tag-vr-community {
        then {
            community add internet-vr;
        }
    }
```

Use the **show policy-options policy-statement *policy-statement-name*** command to display the details of the export policy. The following sample output shows the configuration of the **internet-export-policy** virtual router routing instance. The BGP export policy


Copyright © 2013, Juniper Networks, Inc.

(**internet-export-policy**) is responsible for controlling outbound advertisements to conform with best practices.

```
user@J2350-2-R2> show policy-options policy-statement internet-export-policy
```

```
    term no-leaking {
      from {
        route-filter 0.0.0.0/0 through 0.0.0.0/32 reject;
      }
    }
    term prepend {
      from community prepend-twice;
      then as-path-prepend "65000 65000";
    }
    term adv-cust-routes {
      from community cust-routes;
      then accept;
    }
    term no-transit {
      from as-path null;
      then accept;
    }
    term reject-all {
      then reject;
    }
```

## Virtual Router for the Internet2 Network (VR-i2)

The virtual router routing instance for the Internet2 network is a holding container for the peering sessions and prefixes that are representative of the I2 network.

Use the **show routing-instances *virtual-router-name*** command to display the configuration of the virtual router. The following sample output shows the configuration of the **vr-i2** virtual router routing-instance:

```
user@J2350-2-R2> show routing-instances i2
```

```
  instance-type virtual-router;
  interface ge-0/0/0.210;
  interface ge-0/0/0.280;
  interface lo0.2;
  protocols {
    bgp {
      group i2 {
        type external;
        traceoptions {
          file i2-bgp-trace;
          flag all detail;
        }
        advertise-inactive;
        import i2-import-policy;
        export i2-export-policy;
        peer-as 65222;
        multipath multiple-as;
        neighbor 172.16.0.14;
        neighbor 172.16.0.18;
      }
```

Copyright © 2013, Juniper Networks, Inc.

```
    }
  }
```

Use the **show policy-options policy-statement *policy-statement-name*** command to display the details of the import policy. The following sample output shows the configuration of the **i2-import-policy** policy.

All prefixes learned through the I2 EBGP neighbors are tagged with the community i2 virtual router routing instance (65000:222) by the BGP import policy **i2-import-policy**.

```
user@J2350-2-R2> show policy-options policy-statement i2-import-policy

  term tag-i2-vr-community {
    then {
      community add i2-vr;
    }
  }
```

Use the **show policy-options policy-statement *policy-statement-name*** command to display the details of the export policy. The following sample output shows the configuration of the **i2-export-policy** policy.

The BGP export policy (**i2-export-policy**) is responsible for controlling outbound advertisements to conform with best practices.

```
user@J2350-2-R2> show policy-options policy-statement i2-export-policy

  term no-leaking {
    from {
      route-filter 0.0.0.0/0 through 0.0.0.0/32 reject;
    }
  }
  term adv-cust-routes {
    from community cust-routes;
    then accept;
  }
  term no-transit {
    from as-path null;
    then accept;
  }
  term reject-all {
    then reject;
  }
```

### Virtual Router for the NLR Network (VR-nlr)

The NLR virtual router is a holding container for the peering sessions and prefixes that are representative of the NLR network.

```
user@J2350-2-R2> show routing-instances nlr

  instance-type virtual-router;
  interface ge-0/0/0.200;
  interface lo0.3;
  protocols {
    bgp {
      group nlr {
        type external;
```


Copyright © 2013, Juniper Networks, Inc.

```
        advertise-inactive;
        import nlr-import-policy;
        export nlr-export-policy;
        peer-as 65050;
        multipath multiple-as;
        neighbor 10.0.5.254;
      }
    }
  }
```

Use the **show policy-options policy-statement *policy-statement-name*** command to display the details of the import policy. The following sample output shows the configuration of the **nlr-import-policy** policy.

All prefixes learned through the NLR EBGP neighbors are tagged with the community nlr-vr (65000:333) by the BGP import policy **nlr-import-policy**.

```
user@J2350-2-R2> show policy-options policy-statement nlr-import-policy

  term tag-nlr-vr-community {
    then {
      community add nlr-vr;
    }
  }
```

Use the **show policy-options policy-statement *policy-statement-name*** command to display the details of the export policy. The following sample output shows the configuration of the **nrl-export-policy** policy.

The BGP export policy (**nlr-export-policy**) is responsible for controlling outbound advertisements to conform with best practices.

```
user@J2350-2-R2> show policy-options policy-statement nlr-export-policy

  term no-leaking {
    from {
      route-filter 0.0.0.0/0 through 0.0.0.0/32 reject;
    }
  }
  term adv-cust-routes {
    from community cust-routes;
    then accept;
  }
  term no-transit {
    from as-path null;
    then accept;
  }
  term reject-all {
    then reject;
  }
```

NOTE: For the complete routing table configuration, see "Appendix A - Device Configuration Details" on page 41.

Copyright © 2013, Juniper Networks, Inc.

**Related Documentation**

- Example: Configuring Virtual Routers for Educational Networks on page 12
- Appendix A - Device Configuration Details on page 41

## Example: Configuring Virtual Routers for Educational Networks

This example provides details on configuring virtual router routing instances for network peering using SRX Series or J Series devices as implemented by a network user (Educational Network, in this document called "New University") connected to multiple other networks to exchange traffic.

See "Using Virtual Routers to Provide Customer Peering" on page 1 for overview information.

This topic includes the following sections:



### Requirements

This example uses the following three devices:

- J2350 Services Router as the New University peering device
- EX3200 Ethernet Switch as the service provider network
- SRX210 Services Gateway as the customer device

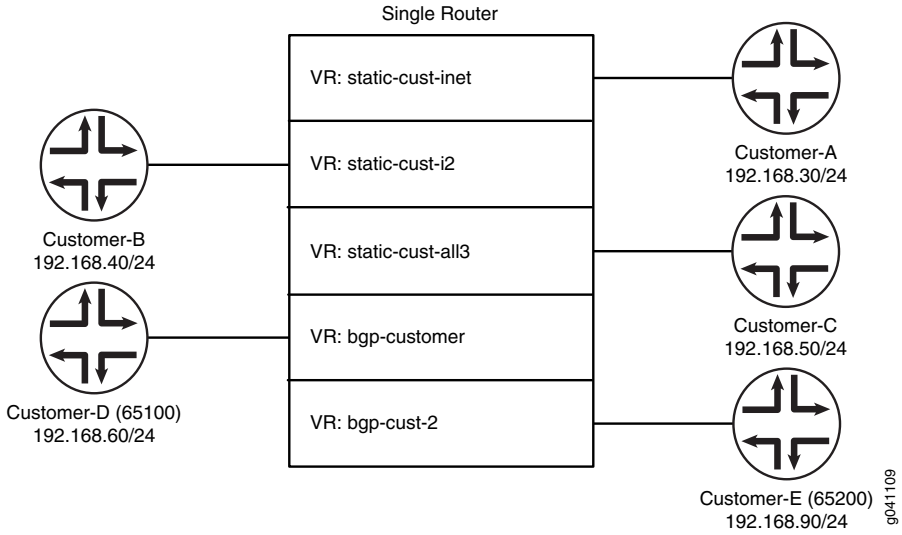
### Network Topology

In this example, New University is connected to multiple other networks to exchange traffic through peering. The example shows a few customer solutions to explain various ways to accommodate statically attached customers of New University and discusses various methods of connecting to customers' devices/networks using BGP.

Figure 4 on page 13 shows the relationship of the virtual router routing instances for customer connectivity. This example explores each of the connectivity models in subsequent sections.


Copyright © 2013, Juniper Networks, Inc.

Figure 4: Virtual Router for Customer Connectivity Topology



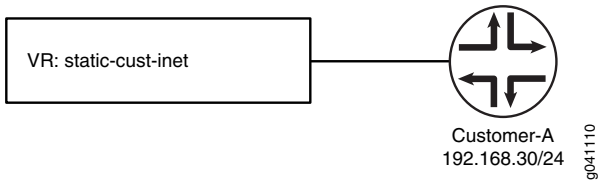

## Example 1: Connecting a Static Customer – Internet Only (Customer A)

New University provides the customer connectivity by advertising the customer's IP address to upstream providers. The users must configure their Internet-facing device to have a default route pointing to New University.

In the example topology, the user (Customer A) network is requesting a simple connection from New University using static rules.

Figure 5 on page 13 shows the topology of the static customer connectivity.

Figure 5: Connecting a Static Customer – Internet Only (Customer A)




This topic includes the following sections:

- Configuring the New University Device on page 14
- Configuring the Customer Device on page 16

Copyright © 2013, Juniper Networks, Inc.

## Configuring the New University Device

**Step-by-Step Procedure**

NOTE: The following configuration includes basic steps to configure the device. For samples of a detailed configuration, see "Appendix A - Device Configuration Details" on page 41.

To configure the New University device:

1. Create a virtual router routing instance (**static-cust-inet**).

   **user@host# set routing-instances static-cust-inet instance-type virtual-router**

2. Assign the **ge-0/0/0.60** interface to the virtual router routing instance.

   **user@host# set routing-instances static-cust-inet interface ge-0/0/0.60**

3. Create the **static-cust-inet-interfaces** RIB group for the interfaces.

   **user@host# set routing-instances static-cust-inet routing-options interface-routes rib-group inet static-cust-inet-interfaces**

   The **static-cust-inet-interfaces** RIB group places the Customer A interfaces into both the **static-cust-inet** virtual router routing instance and the **internet** virtual router routing instance. Looking at the route for 10.0.4.0/30, which is the link between the New University router and the customer network router, you can see that the interface route is now in the routing tables of both virtual routers.

   This is important to maintain reachability for Internet traffic (BGP requires valid next-hops for a route to be active and advertised).

4. Create the **static-cust-inet** RIB group for the static routes.

   **user@host# set routing-instances static-cust-inet routing-options static rib-group inet static-cust-inet**

   The RIB group configuration shares the Customer A networks between the **static-cust-inet** virtual router routing instance and the **internet** virtual router routing instance. These static routes are now available for the **internet** virtual router routing instance to advertise to its upstream providers. To ensure that only New University prefixes are shared upstream, the addition of a BGP community (**cust-routes**) is added to the static routes for later use by the export policy on the Internet peering session. The customer's IP address block is 192.168.30/24.

5. Configure the static routes to be installed in the routing table by specifying the destination of the generated route, and also the next-hop destination (table).

   **user@host# set routing-instances static-cust-inet routing-options static route 0.0.0.0/0 next-table internet.inet.0**

6. Configure the static routes for the routing table by specifying the destination of the generated route, and also the next-hop destination (IP address).

   **user@host# set routing-instances static-cust-all-feeds routing-options static route 192.168.30.0/24 next-hop 10.0.4.1**


Copyright © 2013, Juniper Networks, Inc.

7. Use the **run show route *route-ip-address*** command to display the details for the **10.0.4.0/30** route, which is the link between the New University router and the customer network router.

```
user@J2350-2-R2# run show route 10.0.4.0/30
internet.inet.0: 40 destinations, 63 routes (40 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
10.0.4.0/30 *[Direct/0] 05:05:59
> via ge-0/0/0.60
10.0.4.2/32 *[Local/0] 04:06:47
Local via ge-0/0/0.60
static-cust-inet.inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both
10.0.4.0/30 *[Direct/0] 3d 00:26:15
> via ge-0/0/0.60
10.0.4.2/32 *[Local/0] 3d 00:26:15
Local via ge-0/0/0.60
```

8. Use the **run show route *route-ip-address*** command to display the details for route **192.168.30/24**.

```
user@J2350-2-R2# run show route 192.168.30/24
internet.inet.0: 40 destinations, 63 routes (40 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
192.168.30.0/24 *[Static/5] 04:16:03
> to 10.0.4.1 via ge-0/0/0.60
static-cust-inet.inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both
192.168.30.0/24 *[Static/5] 04:16:03
> to 10.0.4.1 via ge-0/0/0.60
```

Copyright © 2013, Juniper Networks, Inc.

**Results** Use the **show routing-instances *routing-instance-name*** command to display the sample configuration of the New University virtual router routing instances.

```
user@J2350-2-R2> show routing-instances static-cust-inet

  instance-type virtual-router;
  interface ge-0/0/0.60;
  routing-options {
    interface-routes {
      rib-group inet static-cust-inet-interfaces;
    }
    static {
      rib-group static-cust-inet;
      route 0.0.0.0/0 next-table internet.inet.0;
      route 192.168.30.0/24 next-hop 10.0.4.1;
    }
  }
```

Use the **show routing-options** command to display a sample configuration of the RIB groups.

```
user@J2350-2-R2> show routing-options

  rib-groups {
    static-cust-inet {
      import-rib [ static-cust-inet.inet.0 internet.inet.0 ];
      import-policy non-bgp-cust-ribgroup-policy;
    }
    static-cust-inet-interfaces {
      import-rib [ static-cust-inet.inet.0 internet.inet.0 ];
    }
  }
```

Use the **show policy-options policy-statement *policy-name*** command to display a sample configuration of the RIB groups policy. To allow only interfaces to be copied from inet.0, the RIB groups policy is created and applied to the RIB groups.

```
user@J2350-2-R2> show policy-options policy-statement non-bgp-cust-ribgroup-policy

  term reject-default {
    from {
      route-filter 0.0.0.0/0 exact reject;
    }
  }
  term cust-statics {
    from protocol static;
    then {
      community add cust-routes;
      accept;
    }
  }
  term reject-all {
    then reject;
  }
```

## Configuring the Customer Device


Copyright © 2013, Juniper Networks, Inc.

**Step-by-Step Procedure**

NOTE: The following configuration includes basic steps to configure the device. For samples of a detailed configuration, see "Appendix A - Device Configuration Details" on page 41.

To configure the customer device:

1. Create a virtual router routing instance (**cust-a**).

   **user@host# set routing-instances cust-a instance-type virtual-router**

2. Configure the static routes for the routing table by specifying the destination of the generated route, and also the next-hop destination (IP address).

   **user@host# set routing-instances cust-d routing-options static route 0.0.0.0/0 next-hop 10.0.4.2**

3. Use the **run show route *route-ip-address*** command to display the Customer A network details from the ISP routing table.

```
user@EX-3200-1# run show route 192.168.30/24
C1.inet.0: 38 destinations, 70 routes (38 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
192.168.30.0/24 *[BGP/170] 3d 01:43:32, localpref 100
AS path: 65000 I
> to 172.16.0.5 via vlan.230
[BGP/170] 3d 01:43:32, localpref 100
AS path: 65020 65000 I
> to 8.8.8.2 via ge-0/0/14.800
C2.inet.0: 47 destinations, 79 routes (47 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
192.168.30.0/24 *[BGP/170] 3d 01:43:32, localpref 100
AS path: 65000 I
> to 172.16.0.25 via vlan.270
[BGP/170] 3d 01:43:32, localpref 100
AS path: 65010 65000 I
> to 8.8.8.1 via ge-0/0/15.801
```

**Results** Use the **run show routing-options** command to display the sample configuration of the customer device's RIB groups.

```
user@SRX210-A-R3# run show routing-options

static {
  route 0.0.0.0/0 next-hop 10.0.4.2;
}
```

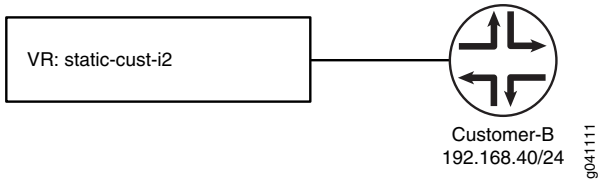
## Example 2: Connecting a Static Customer – Internet2 Only (Customer B)

In this example, the user (Customer B) network is requesting a simple connection from New University along with access to specialized networks such as Internet2 (I2). In this scenario, the customers' connection to New University is placed in the **static-cust-i2** virtual router routing instances, and routing constraints are limited to the virtual router connectivity model.

Figure 6 on page 18 shows the topology of the static customer connectivity.

Copyright © 2013, Juniper Networks, Inc. 

Figure 6: Connecting a Static Customer – Internet2 Only (Customer B)




This topic includes the following section:



## Configuring the New University Device

Step-by-Step Procedure

NOTE: The following configuration includes basic steps to configure the device. For samples of a detailed configuration, see "Appendix A - Device Configuration Details" on page 41.

To configure the New University device:

1. Create a virtual router routing instance (**static-cust-i2**).

   **user@host# set routing-instances static-cust-i2 instance-type virtual-router**

2. Assign the **ge-0/0/0.20** interface to the virtual router routing instance.

   **user@host# set routing-instances static-cust-all-feeds interface ge-0/0/0.20**

3. Create the **static-cust-i2-interfaces** RIB group for the interfaces.

   **user@host# set routing-instances static-cust-all-feeds routing-options interface-routes rib-group inet static-cust-i2-interfaces**

   The RIB group **static-cust-i2-interfaces** places the Customer B interfaces into both the **static-cust-i2** and **internet2** virtual router routing instances. The route 10.0.4.8/30 is the link between the New University router and the customer network routers, and it is available in the routing tables of both virtual routers. This is important to maintain reachability for Internet traffic (BGP requires valid next-hops for a route to be active and advertised).

4. Create the **static-cust-i2** RIB group for the static routes.

   **user@host# set routing-instances static-cust-all-feeds routing-options static rib-group inet static-cust-i2**

   The RIB group **static-cust-i2** shares the Customer B network routes between the **static-cust- inet** virtual router routing instance and the **internet** virtual router routing instance. These static routes are now available for the **internet** virtual router to advertise to its upstream providers. To ensure that only New University prefixes are shared upstream, the addition of a BGP community (**cust-routes**) is added to the static routes for use by the export policy on the Internet peering session. The customer's IP address block is 192.168.40/24.

5. Create an import policy for the instance RIB.


Copyright © 2013, Juniper Networks, Inc.

```
user@host# set routing-instances static-cust-all-feeds routing-options
  instance-import i2-nlr-bgp-instance-import
```

6. Configure the static routes to be installed in the routing table by specifying the destination of the generated route and an option to reach the next-hop to another table.

```
user@host# set routing-instances static-cust-all-feeds routing-options static route
  0.0.0.0/0 next-table i2.inet.0
```

7. Configure the static routes for the routing table by specifying the destination of the generated route, and also the next-hop destination (IP address).

```
user@host# set routing-instances static-cust-all-feeds routing-options static route
  192.168.40.0/24 next-hop 10.0.4.9
```

8. Use the **run show route** ***route-ip-address*** command to display the details for route **10.0.4.8/30**, which is the link between the New University router and the customer network router.

```
user@J2350-2-R2# run show route 10.0.4.8/30
i2.inet.0: 23 destinations, 29 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.0.4.8/30 *[Direct/0] 05:30:07
> via ge-0/0/0.20
10.0.4.10/32 *[Local/0] 04:30:55
Local via ge-0/0/0.20
static-cust-i2.inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both
10.0.4.8/30 *[Direct/0] 3d 00:50:23
> via ge-0/0/0.20
10.0.4.10/32 *[Local/0] 3d 00:50:23
Local via ge-0/0/0.20
```

9. Use the **run show route** ***route-ip-address*** command to display the details for the New University routing table.

```
user@J2350-2-R2# run show route 192.168.40.0/24
i2.inet.0: 23 destinations, 29 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
192.168.40.0/24 *[Static/5] 04:31:31
> to 10.0.4.9 via ge-0/0/0.20
static-cust-i2.inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both
192.168.40.0/24 *[Static/5] 04:31:31
> to 10.0.4.9 via ge-0/0/0.20
```

10. Use the **run show route** ***route-ip-address*** command to display the Customer B network details from the ISP routing table.

```
user@EX-3200-1# run show route 192.168.40/24
T1.inet.0: 22 destinations, 39 routes (22 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
192.168.40.0/24 *[BGP/170] 3d 00:53:37, localpref 100
AS path: 65000 I
> to 172.16.0.13 via vlan.210
[BGP/170] 3d 00:40:18, localpref 100, from 130.130.2.2
AS path: 65000 I
> to 172.16.0.13 via vlan.210
```

Copyright © 2013, Juniper Networks, Inc.

```
T2.inet.0: 23 destinations, 40 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
192.168.40.0/24 *[BGP/170] 3d 00:40:18, localpref 100
AS path: 65000 I
> to 172.16.0.17 via vlan.280
[BGP/170] 3d 00:53:37, localpref 100, from 130.130.1.1
AS path: 65000 I
> to 172.16.0.17 via vlan.280
```

 Copyright © 2013, Juniper Networks, Inc.

**Results** Use the **show routing-instances *routing-instance-name*** command to display the sample configuration of New University virtual router routing instances.

```
user@J2350-2-R2> show routing-instances static-cust-i2

  instance-type virtual-router;
  interface ge-0/0/0.20;
  routing-options {
    interface-routes {
      rib-group inet static-cust-i2-interfaces;
    }
    static {
      rib-group static-cust-i2;
      route 0.0.0.0/0 next-table i2.inet.0;
      route 192.168.40.0/24 next-hop 10.0.4.9;
    }
  }
```

Use the **show routing-options** command to display the configuration of the RIB groups.

```
user@J2350-2-R2> show routing-options

  rib-groups {
    static-cust-i2 {
      import-rib [ static-cust-i2.inet.0 i2.inet.0 ];
      import-policy non-bgp-cust-ribgroup-policy;
    }
    static-cust-i2-interfaces {
      import-rib [ static-cust-i2.inet.0 i2.inet.0 ];
    }
  }
```

Use the **show policy-options policy-statement *policy-name*** command to display the configuration of the RIB groups policy.

```
user@J2350-2-R2> show policy-options policy-statement non-bgp-custribgroup-policy

  term reject-default {
    from {
      route-filter 0.0.0.0/0 exact reject;
    }
  }
  term cust-statics {
    from protocol static;
    then {
      community add cust-routes;
      accept;
    }
  }
  term reject-all {
    then reject;
  }
```

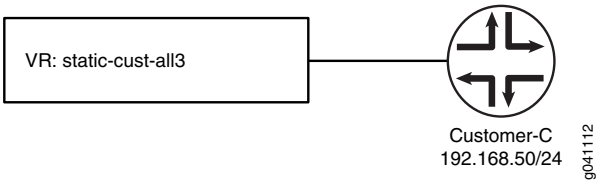
## Example 3: Connecting a Static Customer – Access to All Three Networks (Customer C)

In this example, the user (Customer C) network is requesting multiple connections simultaneously from New University. The user is depending on New University to provide

the best route to each destination based on New University routing policies.

Figure 7 on page 22 shows the topology of the static customer connectivity.

Figure 7: Connecting a Static Customer – Access to Three Different Networks (Customer C)




In this example, a static route to the **internet.inet.0** table is maintained, but additional prefixes are imported in the virtual router's routing table from **i2.inet.0** and **nlr.inet.0**. This configuration reduces the size of the routing table, and the route is preferred for the specialized networks to the traditional Internet.

This topic includes the following section:

- Configuring the New University Device on page 22

## Configuring the New University Device

**Step-by-Step Procedure**

NOTE: The following configuration includes basic steps to configure the device. For samples of a detailed configuration, see "Appendix A - Device Configuration Details" on page 41.

To configure the New University device:

1. Create a virtual router routing instance (**static-cust-all-feeds**).

   **user@host# set routing-instances static-cust-all-feeds instance-type virtual-router**

2. Assign the **ge-0/0/0.10** interface to the virtual router routing instance.

   **user@host# set routing-instances static-cust-all-feeds interface ge-0/0/0.10**

3. Create the **static-cust-all-feeds-interfaces** RIB group for the interfaces.

   **user@host# set routing-instances static-cust-all-feeds routing-options interface-routes rib-group inet static-cust-all-feeds-interfaces**

4. Create the **static-cust-all-feeds** RIB group for the static routes.

   **user@host# set routing-instances static-cust-all-feeds routing-options static rib-group inet static-cust-all-feeds**

5. Create an import policy for the instance RIB.

   **user@host# set routing-instances static-cust-all-feeds routing-options instance-import i2-nlr-bgp-instance-import**

6. Configure the static routes for the routing table by specifying the destination of the generated route, and also the next-hop destination (table).


Copyright © 2013, Juniper Networks, Inc.

```
user@host# set routing-instances static-cust-all-feeds routing-options static route
  0.0.0.0/0 next-table internet.inet.0
```

7. Configure the static routes for the routing table by specifying the destination of the generated route, and also the next-hop destination (IP address).

```
user@host# set routing-instances static-cust-all-feeds routing-options static route
  192.168.50.0 next-hop 10.0.4.13
```

8. Use the **run show route table** *route-table-name* command to display the details for the routing table.

```
user@J2350-2-R2# run show route table static-cust-all-feeds
static-cust-all-feeds.inet.0: 15 destinations, 15 routes (15 active, 0
holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
0.0.0.0/0 *[Static/5] 05:01:00
to table internet.inet.0
10.0.4.12/30 *[Direct/0] 3d 01:20:28
> via ge-0/0/0.10
10.0.4.14/32 *[Local/0] 3d 01:20:28
Local via ge-0/0/0.10
10.10.1.0/24 *[BGP/170] 3d 01:20:28, localpref 110, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
10.10.2.0/24 *[BGP/170] 3d 01:20:28, localpref 110, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
10.10.3.0/24 *[BGP/170] 3d 01:20:28, localpref 110, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
10.10.4.0/24 *[BGP/170] 3d 01:20:28, localpref 110
AS path: 65222 I
> to 172.16.0.14 via ge-0/0/0.210
to 172.16.0.18 via ge-0/0/0.280
10.10.5.0/24 *[BGP/170] 3d 01:20:28, localpref 110, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
10.10.11.0/24 *[BGP/170] 05:07:15, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
10.10.12.0/24 *[BGP/170] 05:07:15, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
10.10.13.0/24 *[BGP/170] 05:07:15, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
10.10.14.0/24 *[BGP/170] 05:07:15, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
120.120.0.0/16 *[BGP/170] 05:07:15, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
130.130.0.0/16 *[BGP/170] 3d 01:20:28, localpref 110, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
```

Copyright © 2013, Juniper Networks, Inc.

```
192.168.50.0/24 *[Static/5] 05:01:00
> to 10.0.4.13 via ge-0/0/0.10
```


Copyright © 2013, Juniper Networks, Inc.

**Results** Use the **show routing-instances *routing-instance-name*** command to display the sample configuration of New University virtual router routing instances.

```
user@J2350-2-R2> show routing-instances static-cust-all-feeds
```

```
  instance-type virtual-router;
  interface ge-0/0/0.10;
  routing-options {
    interface-routes {
      rib-group inet static-cust-all-feeds-interfaces;
    }
    static {
      rib-group static-cust-all-feeds;
      route 0.0.0.0/0 next-table internet.inet.0;
      route 192.168.50.0/24 next-hop 10.0.4.13;
    }
    instance-import i2-nlr-bgp-instance-import;
  }
```

Use the **show policy-options policy-statement *policy-name*** command to display a sample configuration of the RIB groups policy.

```
user@J2350-2-R2> show policy-options policy-statement i2-nlr-bgp-instanceimport
```

```
  term accept-i2 {
    from {
      instance i2;
      community i2-vr;
    }
    then {
      local-preference 110;
      accept;
    }
  }
  term accept-nlr {
    from {
      instance nlr;
      community nlr-vr;
    }
    then accept;
  }
  term reject-all {
    then reject;
  }
```

Copyright © 2013, Juniper Networks, Inc.

NOTE:

When you configure the policy, consider the following:

- Configure the policy statement to define matching criteria and the actions to be taken for traffic that matches the criteria. The use of the BGP community ensures that only those routes learned from upstream peers, and not other New University peers, are accepted into the **static-cust-all-feeds** routing table.
- Import the **i2.inet.0** and the **nlr.inet.0** routes into the **static-cust-all-feeds** virtual router routing instances, and configure this policy under the instance-import option of the default routing instance.

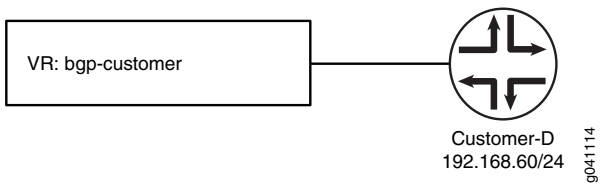
NOTE: Only the specific prefixes for the I2 and NLR networks are populated in the routing table. For all other destinations, the default route points to the **internet.inet.0** virtual router routing table.

## Example 4: BGP Customer – Creating a Single Peering Session (Customer D)

In this example, the user (Customer D) network is requesting multiple connections simultaneously from New University and to configure some of the routing-related options themselves.

Figure 8 on page 26 shows the topology of the static customer connectivity.

### Figure 8: BGP Customer – Creating a Single Peering Session (Customer D)




In this configuration, a single BGP peering session is provided to the customer where the customer can configure part of it. However, configurations related to route preference and control over active prefixes for the customer connection are controlled and maintained by New University.

This topic includes the following sections:




Copyright © 2013, Juniper Networks, Inc.

## Configuring the New University Device

**Step-by-Step Procedure**

NOTE: The following configuration includes basic steps to configure the device. For samples of a detailed configuration, see "Appendix A - Device Configuration Details" on page 41.

To configure the New University device:

1. Create a virtual router routing instance (**bgp-customer**).

   **user@host# set routing-instances bgp-customer instance-type virtual-router**

2. Assign the **ge-0/0/0.30** interface to the virtual router routing instance.

   **user@host# set routing-instances bgp-cust-2 interface ge-0/0/0.30**

3. Create the **bgp-customer-rg-interfaces** RIB group for the interfaces.

   **user@host# set routing-instances bgp-customer routing-options interface-routes rib-group inet bgp-customer-rg-interfaces**

4. Create the **bgp-customers-rg** RIB group for the static routes.

   **user@host# set routing-instances bgp-customer routing-options static rib-group inet bgp-customers-rg**

5. Create an import policy for the instance RIB.

   **user@host# set routing-instances bgp-customer routing-options instance-import bgp-cust-instance-import**

6. Configure the family type to be unicast.

   **user@host# set routing-instances bgp-customer protocols bgp family inet unicast rib-group bgp-customers-rg**

   This configuration is required to enable the BGP peers to carry the unicast routes that are being used for unicast forwarding purposes.

7. Set the BGP group customer type to **external**.

   **user@host# set routing-instances bgp-customer protocols bgp group customer type external**

8. Configure a BGP neighbor (peer).

   **user@host# set routing-instances bgp-customer protocols bgp group customer neighbor 10.0.4.5 peer-as 65100**

9. Configure the **cust-d-export** BGP group export policy.

   **user@host# set routing-instances bgp-customer protocols bgp group customer neighbor 10.0.4.5 export cust-d-export**

Copyright © 2013, Juniper Networks, Inc.

**Results** Use the **show routing-instances *routing-instance-name*** command to display the sample configuration of New University virtual router routing instances.

```
user@J2350-2-R2> show routing-instances bgp-customer

  instance-type virtual-router;
  interface ge-0/0/0.30;
  routing-options {
    interface-routes {
      rib-group inet bgp-customer-rg-interfaces;
    }
    static {
      rib-group bgp-customers-rg;
    }
    instance-import bgp-cust-instance-import;
  }
  protocols {
    bgp {
      family inet {
        unicast {
          rib-group bgp-customers-rg;
        }
      }
      group customers {
        type external;
        advertise-inactive;
        neighbor 10.0.4.5 {
          export cust-d-export;
          peer-as 65100;
        }
      }
    }
  }
```

NOTE: To allow only interfaces to be copied from inet.0, you must configure a policy and apply it to the RIB group.

Use the **show policy-options policy-statement *policy-name*** command to display the sample configuration of the RIB groups policy.

```
user@J2350-2-R2> show policy-options policy-statement bgp-cust-instanceimport

  term nlr {
    from {
      instance nlr;
      community nlr-vr;
    }
    then {
      local-preference 105;
      accept;
    }
  }
    term internet {
      from {
```


Copyright © 2013, Juniper Networks, Inc.

```
        instance internet;
      community internet-vr;
    }
    then accept;
  }
  term i2 {
    from {
      instance i2;
      community i2-vr;
    }
    then {
      local-preference 110;
      accept;
    }
  }
  term reject-all {
    then reject;
  }
```

NOTE:

In the previous policy configuration:

- Using the instance-import policy, the local preference is configured to meet the requirements of the downstream customers. The customers do not see the change in the local-preference, but those prefixes that are shared with the customer are the only active prefixes available to the customer. If a failure occurs that limits connectivity to one of the specialized networks, then the standard routes become active and are advertised to the customer router.
- The use of BGP allows customers to peer with New University in a single session. An additional RIB group is created to ensure that Customer D's networks are propagated to all three upstream networks.

Use the **show routing-instances *routing-instance-name* protocols bgp** command to display the sample configuration of the BGP group policy.

```
user@J2350-2-R2> show routing-instances bgp-customer protocols bgp
  family inet {
    unicast {
      rib-group bgp-customers-rg;
    }
  }
  group customers {
    type external;
    advertise-inactive;
    neighbor 10.0.4.5 {
      export cust-d-export;
      peer-as 65100;
    }
  }
```

Copyright © 2013, Juniper Networks, Inc.

NOTE: In the previous configuration, BGP is configured in the standard way. One benefit of the hierarchical configuration is that a standard peer-group can be used to handle many distinct customers. By placing the peer-specific information under the neighbor statement, the need to create individual groups is reduced.

Use the **show routing-options rib-groups *rib-groups-name*** command to display the sample configuration of the RIB groups.

```
user@J2350-2-R2> show routing-options rib-groups bgp-customers-rg

  import-rib [ bgp-customer.inet.0 internet.inet.0 i2.inet.0 nlr.inet.0 ];
  import-policy static-all-feeds-cust-ribgroup-policy;
```

NOTE: In the previous configuration, the **internet.inet.0**, **i2.inet.0**, and the **nlr.inet.0.** routes are imported into the **inet bgp-customer-rg-interfaces** virtual router routing instance, and this policy is configured under the instance-import option of the default routing instance. This configuration combines three provider networks.

Use the **show policy-options policy-statement *policy-name*** command to verify the instance-import policy.

```
user@J2350-2-R2> show policy-options policy-statement static-all-feeds-cust-ribgroup-policy

  term reject-default {
    from {
      route-filter 0.0.0.0/0 exact reject;
    }
  }
  term cust-statics {
    from protocol static;
    then {
      community add cust-routes;
      community add prepend-twice;
      accept;
    }
  }
  term cust-bgp {
    from protocol bgp;
    then {
      community add cust-routes;
      community add prepend-twice;
      accept;
    }
  }
  term reject-all {
    then reject;
  }
```


Copyright © 2013, Juniper Networks, Inc.

NOTE: The RIB group and associated policy have additional configuration to enable prepending of prefixes. Prepending is used only for the Internet virtual router. Since most New University networks peer to the Internet and I2, and customers prefer the use of the faster Internet2 network. Prepending a prefix with two additional AS-PATH attributes on the Internet links ensures that those prefixes are preferred through the Internet2 peering sessions (shorter AS-PATH).

## Configuring the Customer Device

**Step-by-Step Procedure**

NOTE: The following configuration includes basic steps to configure the device. For samples of a detailed configuration, see "Appendix A - Device Configuration Details" on page 41.

To configure the customer device:

1. Create a virtual router routing instance (**cust-d**).

   **user@host# set routing-instances cust-d instance-type virtual-router**

2. Assign the **fe-0/0/0.30** interface to the virtual router routing instance.

   **user@host# set routing-instances cust-d interface fe-0/0/0.30**

3. Assign the **lo0.30** interface to the virtual router routing instance.

   **user@host# set routing-instances cust-d interface lo0.30**

4. Configure the static routes to be installed in the routing table by specifying the destination of the generated route, and the destination where packets should not be forwarded (**discard**).

   **user@host# set routing-instances cust-d routing-options static route 192.168.60.0/24 discard**

5. Specify the device's AS number as assigned by the Network Information Center (NIC) in the United States.

   **user@host# set routing-instances cust-d routing-options autonomous-system 65100**

6. Apply one or more policies to routes being exported from the routing table into a routing protocol.

   **user@host# set routing-instances cust-d protocols bgp group provider-ebgp export cust-d-export**

7. Configure a BGP neighbor (peer).

   **user@host# set routing-instances cust-d protocols bgp group provider-ebgp peer-as 65000**

8. Configure a BGP neighbor (peer) with IP address **10.0.4.6**.

Copyright © 2013, Juniper Networks, Inc.

```
user@host# set routing-instances cust-d protocols bgp group provider-ebgp neighbor
  10.0.4.6
```

**Results** Use the **show routing-instances *routing-instance-name*** command to display the complete configuration. The sample output in this example is truncated to provide only the details relevant to the virtual routers configuration.

```
user@SRX210-A-R3> show routing-instances cust-d

  instance-type virtual-router;
  interface fe-0/0/0.30;
  interface lo0.30;
  routing-options {
    static {
      route 192.168.60.0/24 discard;
    }
    autonomous-system 65100;
  }
  protocols {
    bgp {
      group provider-ebgp {
        type external;
        export cust-d-export;
        peer-as 65000;
        neighbor 10.0.4.6;
      }
    }
  }
```

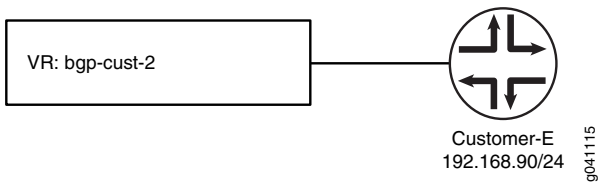
NOTE: The only disadvantage of this solution is that the upstream prefixes are replicated between several virtual router routing tables.

## Example 5: BGP Customer – Creating Multiple Peering Sessions (Customer E)

In this example, the BGP user (Customer E) network is requesting multiple connections from New University along with configuration privileges. This configuration requires a single PE-CE link and multiple BGP peering sessions for each upstream provider feed using multihop external BGP (EBGP).

Figure 9 on page 32 shows the topology of the static customer connectivity.

### Figure 9: BGP Customer – Creating Multiple Peering Sessions (Customer E)





Copyright © 2013, Juniper Networks, Inc.

In this example, each group contains a BGP group for customers. These customer peering statements must use the EBGP multihop option since they are peering to a loopback address in each of the virtual router routing instances. You must also add the interface routes to a RIB group for each instance. This configuration propagates the loopback interfaces to the **bgp-cust-2** virtual router, so that it can form a BGP peering session with the customer.

This section contains the following procedures:



## Configuring the New University Device

**Step-by-Step Procedure**

NOTE: The following configuration includes basic steps to configure the device. For samples of a detailed configuration, see "Appendix A - Device Configuration Details" on page 41.

In this configuration, the virtual router routing instances provide multiple BGP sessions for each network. The primary BGP connectivity is contained within each provider of the virtual router.

To configure the New University virtual router routing instance:

1. Create a virtual router routing instance (**bgp-cust-2**).

   **user@host# set routing-instances bgp-cust-2 instance-type virtual-router**

2. Assign the **bgp-cust-2** interface to the virtual router routing instance.

   **user@host# set routing-instances bgp-cust-2 interface ge-0/0/0.40**

3. Create a RIB group for the interfaces.

   **user@host# set routing-instances bgp-cust-2 routing-options interface-routes rib-group inet bgp-cust2-interfaces**

## Configuring the New University Device for the Internet Virtual Router Routing Instance

**Step-by-Step Procedure**

To configure the New University virtual router for the **internet** virtual router routing instance:

1. Create a virtual router routing instance (**internet**).

   **user@host# set routing-instances internet instance-type virtual-router**

Copyright © 2013, Juniper Networks, Inc.

2. Assign the **ge-0/0/0.230** interface to the virtual router routing instance.

   **user@host# set routing-instances bgp-cust-2 interface ge-0/0/0.230**

3. Assign the **ge-0/0/0.270** interface to the virtual router routing instance.

   **user@host# set routing-instances bgp-cust-2 interface ge-0/0/0.270**

4. Assign the **lo0.2** interface to the virtual router routing instance.

   **user@host# set routing-instances bgp-cust-2 interface lo0.2**

5. Create the **bgp-cust2-inet-interfaces** RIB group for the interfaces.

   **user@host# set routing-instances internet routing-options interface-routes rib-group inet bgp-cust2-inet-interfaces**

6. Configure the BGP group customer type as **external**.

   **user@host# set routing-instances internet protocols bgp group customers type external**

7. Configure the BGP group multihop with a time-to-live (TTL) value of **2**.

   This enables external peering session, which allows unconnected third-party next-hops.

   **user@host# set routing-instances internet protocols bgp multihop ttl 2**

8. Specify the address of the local end of a BGP session as **1.1.1.1**.

   **user@host# set routing-instances internet protocols bgp local-address 1.1.1.1**

9. Configure the BGP group by including the **advertise-inactive** statement.

   The BGP advertises the best route even if the routing table does not select it to be the active route.

   **user@host# set routing-instances internet protocols bgp advertise-inactive**

10. Configure the BGP group policy **cust-routes-import**.

    This configuration applies to one or more routing policies to the routes being imported into the routing table from BGP.

    **user@host# set routing-instances internet protocols bgp import cust-routes-import**

11. Configure a BGP neighbor (peer) with the IP address **10.0.2.5**, and configure the peer as **65200**.

    **user@host# set routing-instances internet protocols bgp neighbor 10.0.2.5 peer-as 65200**

### Configuring the New University Device for the Internet2 Virtual Router Routing Instance

**Step-by-Step Procedure**

To configure the New University virtual router routing instance for the Internet2 connection:

1. Create a virtual router routing instance (**i2** ).

   **user@host# set routing-instances i2 instance-type virtual-router**

2. Assign the **ge-0/0/0.210** interface to the virtual router routing instance.


Copyright © 2013, Juniper Networks, Inc.

```
user@host# set routing-instances bgp-cust-2 interface ge-0/0/0.210
```

3. Assign the **ge-0/0/0.280** interface to the virtual router routing instance.

```
user@host# set routing-instances bgp-cust-2 interface ge-0/0/0.280
```

4. Assign the **lo0.2** interface to the virtual router routing instance.

```
user@host# set routing-instances bgp-cust-2 interface lo0.2
```

5. Create the **bgp-cust2-i2-interfaces** RIB group for the interfaces.

```
user@host# set routing-instances i2 routing-options interface-routes rib-group inet
  bgp-cust2-i2-interfaces
```

6. Configure the BGP group customer type as **external**.

```
user@host# set routing-instances i2 protocols bgp group customers type external
```

7. Configure the BGP group multihop with a time-to-live (TTL) value of **2**.

   This enables external peering session, which allows unconnected third-party next-hops.

```
user@host# set routing-instances i2 protocols bgp multihop ttl 2
```

8. Specify the address of the local end of a BGP session as **1.1.1.2**.

```
user@host# set routing-instances i2 protocols bgp local-address 1.1.1.2
```

9. Configure the BGP group by including the **advertise-inactive** statement.

   The BGP advertises the best route even if the routing table does not select it to be the active route.

```
user@host# set routing-instances i2 protocols bgp advertise-inactive
```

10. Configure the **cust-routes-import** BGP group policy.

    This configuration applies to one or more routing policies to the routes being imported into the routing table from BGP.

```
user@host# set routing-instances i2 protocols bgp import cust-routes-import
```

11. Configure a BGP neighbor (peer) with an IP address of **10.0.2.5**, and a peer autonomous system number of **65200**.

```
user@host# set routing-instances i2 protocols bgp neighbor 10.0.2.5 peer-as 65200
```

### Configuring the New University Device for the NLR Virtual Router Routing Instance

**Step-by-Step Procedure**

To configure the New University virtual router routing instance for the NLR connection:

1. Create a virtual router routing instance (**nlr**).

```
user@host# set routing-instances nlr instance-type virtual-router
```

2. Assign the **ge-0/0/0.200** interface to the virtual router routing instance.

```
user@host# set routing-instances bgp-cust-2 interface ge-0/0/0.200
```

3. Assign the **lo0.3** interface to the virtual router routing instance.

```
user@host# set routing-instances bgp-cust-2 interface lo0.3
```

Copyright © 2013, Juniper Networks, Inc.

4. Create the **bgp-cust2-nlr-interfaces** RIB group for the interfaces.

   **user@host# set routing-instances nlr routing-options interface-routes rib-group inet bgp-cust2-nlr-interfaces**

5. Configure the BGP group customer type as **external**.

   **user@host# set routing-instances nlr protocols bgp group customers type external**

6. Configure the BGP group multihop with a time-to-live (TTL) value of **2**.

   This enables external peering session, which allows unconnected third-party next-hops.

   **user@host# set routing-instances nlr protocols bgp multihop ttl 2**

7. Specify the address of the local end of a BGP session as **1.1.1.3**.

   **user@host# set routing-instances nlr protocols bgp local-address 1.1.1.3**

8. Configure the BGP group by including the **advertise-inactive** statement.

   The BGP advertises the best route even if the routing table does not select it to be the active route.

   **user@host# set routing-instances nlr protocols bgp advertise-inactive**

9. Configure the BGP group policy **cust-routes-import**.

   This configuration applies to one or more routing policies to the routes being imported into the routing table from BGP.

   **user@host# set routing-instances nlr protocols bgp import cust-routes-import**

10. Configure a BGP neighbor (peer) with IP address **10.0.2.5**, and configure the peer as **65200**.

    **user@host# set routing-instances nlr protocols bgp neighbor 10.0.2.5 peer-as 65200**

11. Use the **run show route table *route-ip-address-table-name* terse** command to display the details of the routing table.

    **user@J2350-2-R2# run show route table bgp-cust-2 terse**

```
bgp-cust-2.inet.0: 15 destinations, 15 routes (15 active, 0 holddown, 0
hidden)
+ = Active Route, - = Last Active, * = Both
A Destination P Prf Metric 1 Metric 2 Next hop AS path
* 1.1.1.1/32 D 0 >lo0.1
* 1.1.1.2/32 D 0 >lo0.2
* 1.1.1.3/32 D 0 >lo0.3
* 10.0.2.4/30 D 0 >ge-0/0/0.40
* 10.0.2.6/32 L 0 Local
* 10.0.5.0/24 D 0 >ge-0/0/0.200
* 10.0.5.2/32 L 0 Local
* 172.16.0.4/30 D 0 >ge-0/0/0.230
* 172.16.0.5/32 L 0 Local
* 172.16.0.12/30 D 0 >ge-0/0/0.210
* 172.16.0.13/32 L 0 Local
* 172.16.0.16/30 D 0 >ge-0/0/0.280
* 172.16.0.17/32 L 0 Local
* 172.16.0.24/30 D 0 >ge-0/0/0.270
* 172.16.0.25/32 L 0 Local
```

 Copyright © 2013, Juniper Networks, Inc.

**Results** Use the **show routing-instances** ***routing-instance-name*** command to display the complete configuration. The sample output is truncated to provide configuration details relevant to the virtual router routing instances.

```
user@J2350-2-R2> show routing-instances bgp-cust-2

  instance-type virtual-router;
  interface ge-0/0/0.40;
  routing-options {
    interface-routes {
      rib-group inet bgp-cust2-interfaces;
    }
  }
  i2 {
    instance-type virtual-router;
    interface ge-0/0/0.210;
    interface ge-0/0/0.280;
    interface lo0.2;
    routing-options {
      interface-routes {
        rib-group inet bgp-cust2-i2-interfaces;
      }
    }
    protocols {
      bgp {
        group customers {
          type external;
          multihop {
            ttl 2;
          }
          local-address 1.1.1.2;
          advertise-inactive;
          import cust-routes-import;
          neighbor 10.0.2.5 {
            peer-as 65200;
          }
        }
      }
    }
  }
  internet {
    instance-type virtual-router;
    interface ge-0/0/0.230;
    interface ge-0/0/0.270;
    interface lo0.1;
    routing-options {
      interface-routes {
        rib-group inet bgp-cust2-inet-interfaces;
      }
    }
    protocols {
      bgp {
        group customers {
          type external;
          multihop {
            ttl 2;
```

Copyright © 2013, Juniper Networks, Inc.

```
            }
            local-address 1.1.1.1;
            advertise-inactive;
            import cust-routes-import;
            neighbor 10.0.2.5 {
              peer-as 65200;
            }
          }
        }
      }
    }
    nlr {
      instance-type virtual-router;
      interface ge-0/0/0.200;
      interface lo0.3;
      routing-options {
        interface-routes {
          rib-group inet bgp-cust2-nlr-interfaces;
        }
      }
      protocols {
        bgp {
          group customers {
            type external;
            multihop {
              ttl 2;
            }
            local-address 1.1.1.3;
            advertise-inactive;
            import cust-routes-import;
            neighbor 10.0.2.5 {
              peer-as 65200;
            }
          }
        }
      }
    }
```

Use the **show routing-options rib-groups *rib-groups-name*** command to verify the RIB group configurations.

```
user@J2350-2-R2> show routing-options rib-groups bgp-cust2-interfaces
import-rib [ bgp-cust-2.inet.0 internet.inet.0 i2.inet.0 nlr.inet.0 ];

user@J2350-2-R2> show routing-options rib-groups bgp-cust2-i2-interfaces
import-rib [ i2.inet.0 bgp-cust-2.inet.0 ];

user@J2350-2-R2> show routing-options rib-groups bgp-cust2-nlr-interfaces
import-rib [ nlr.inet.0 bgp-cust-2.inet.0 ];
```

### Configuring the Customer Device

**Step-by-Step Procedure**

To configure the customer device:

1. Create a virtual router routing instance (**cust-e**).


Copyright © 2013, Juniper Networks, Inc.

```
user@host# set routing-instances cust-e instance-type virtual-router
```

2. Assign the **fe-0/0/0.40** interface to the virtual router routing instance.

```
user@host# set routing-instances bgp-cust-2 interface fe-0/0/0.40
```

3. Configure the static routes for the routing table by specifying the destination of the generated route, and also the next-hop destination (IP address).

```
user@host# set routing-instances cust-e routing-options static route 1.1.1.0/29
  next-hop 10.0.2.6
```

4. Configure the static routes to be installed in the routing table by specifying the destination of the generated route, and the destination where packets should not be forwarded (**discard**).

```
user@host# set routing-instances cust-e routing-options static route
  192.168.90.0/24 discard
```

5. Specify the device's AS number as assigned by the Network Information Center (NIC) in the United States.

```
user@host# set routing-instances cust-e routing-options autonomous-system
  65200
```

6. Configure an external BGP group (**provider-ebgp**).

```
user@host# set routing-instances cust-e protocols bgp group provider-ebgp type
  external
```

7. Configure the BGP group multihop with a time-to-live (TTL) value of **2**.

   This enables external peering session, which allows unconnected third-party next-hops.

```
user@host# set routing-instances cust-e protocols bgp group provider-ebgp type
  external multihop ttl 2
```

8. Apply one or more policies to routes being exported from the routing table into a routing protocol.

```
user@host# set routing-instances cust-e protocols bgp group provider-ebgp export
  cust-e-export
```

9. Configure a BGP neighbor (peer).

```
user@host# set routing-instances cust-e protocols bgp group provider-ebgp peer-as
  65000
```

10. Configure a BGP neighbor (peer) with IP address **1.1.1.1**.

```
user@host# set routing-instances cust-e protocols bgp group provider-ebgp neighbor
  1.1.1.1
```

11. Configure a BGP neighbor (peer) with IP address **1.1.1.2**.

```
user@host# set routing-instances cust-e protocols bgp group provider-ebgp neighbor
  1.1.1.2
```

12. Configure a BGP neighbor (peer) with IP address **1.1.1.3**.

Copyright © 2013, Juniper Networks, Inc.

```
user@host# set routing-instances cust-e protocols bgp group provider-ebgp neighbor
  1.1.1.3
```

13. Use the **run show route table *route-table-name* terse** command to verify the routing table (customer device).

```
user@SRX210-A-R3# run show route table cust-e terse
cust-e.inet.0: 30 destinations, 41 routes (30 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
A Destination P Prf Metric 1 Metric 2 Next hop AS path
* 1.1.1.0/29 S 5 >10.0.2.6
* 10.0.2.4/30 D 0 >fe-0/0/0.40
* 10.0.2.5/32 L 0 Local
* 10.0.3.4/32 B 170 100 >10.0.2.6 65000 65020 65010 I
* 10.10.1.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65222 I
* 10.10.2.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65222 I
* 10.10.3.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65222 I
* 10.10.4.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65222 I
* 10.10.5.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65222 I
* 10.10.6.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.7.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.8.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.9.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.10.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.11.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65050 I
* 10.10.12.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65050 I
* 10.10.13.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65050 I
* 10.10.14.0/24 B 170 100 >10.0.2.6 65000 65010 I
B 170 100 >10.0.2.6 65000 65050 I
* 10.10.15.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.16.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.17.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.18.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.19.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 10.10.20.0/24 B 170 100 >10.0.2.6 65000 65010 I
* 120.120.0.0/16 B 170 100 >10.0.2.6 65000 65050 I
* 130.130.0.0/16 B 170 100 >10.0.2.6 65000 65222 I
* 192.168.60.0/24 B 170 100 >10.0.2.6 65000 65100 I
B 170 100 >10.0.2.6 65000 65100 I
B 170 100 >10.0.2.6 65000 65100 I
* 192.168.90.0/24 S 5 Discard
* 200.200.0.0/16 B 170 100 >10.0.2.6 65000 65010 I
* 201.201.0.0/16 B 170 100 >10.0.2.6 65000 65020 I
```

 Copyright © 2013, Juniper Networks, Inc.

**Results** Use the **show routing-instances** *routing-instance-name* command to verify customer device configuration.

```
user@SRX210-A-R3> show routing-instances cust-e

  instance-type virtual-router;
  interface fe-0/0/0.40;
  routing-options {
    static {
      route 1.1.1.0/29 next-hop 10.0.2.6;
      route 192.168.90.0/24 discard;
    }
    autonomous-system 65200;
  }
  protocols {
    bgp {
      group provider-ebgp {
        type external;
        multihop {
          ttl 2;
        }
        export cust-e-export;
        peer-as 65000;
        neighbor 1.1.1.1;
        neighbor 1.1.1.2;
        neighbor 1.1.1.3;
      }
    }
  }
```

**Related Documentation**

- Using Virtual Routers to Provide Customer Peering on page 1
- Appendix A - Device Configuration Details on page 41

## Appendix A - Device Configuration Details

This appendix includes the following topics:



### Educational Network Device Configuration

The "Example: Configuring Virtual Routers for Educational Networks" on page 12 example uses a J2350 Services Router as the Educational Network (New University) device.

Use the **show** command to display the configuration of this device.

```
user@J2350-2-R2> show | no-more

  ## Last changed: 2010-04-05 09:29:36 EDT
  version 9.3R3.8;
```

Copyright © 2013, Juniper Networks, Inc.

```
system {
    host-name J2350-2-R2;
    time-zone America/New_York;
    root-authentication {
        encrypted-password "$1 "; ## SECRET-DATA
    }
    name-server {
        68.87.68.162;
        68.87.74.162;
    }
    login {
        user user {
            uid 2000;
            class super-user;
            authentication {
                encrypted-password "$1"; ## SECRET-DATA
            }
        }
        user trusteduser {
            uid 2002;
            class super-user;
            authentication {
                encrypted-password "$1"; ## SECRET-DATA
            }
        }
    }
    services {
        ftp;
        ssh;
        telnet;
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any any;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
    ntp {
        server 18.26.4.105;
    }
}
interfaces {
    ge-0/0/0 {
        vlan-tagging;
        unit 10 {
            vlan-id 10;
            family inet {
                address 10.0.4.14/30;
            }
        }
```


Copyright © 2013, Juniper Networks, Inc.

```
unit 20 {
    vlan-id 20;
    family inet {
        address 10.0.4.10/30;
    }
}
unit 30 {
    vlan-id 30;
    family inet {
        address 10.0.4.6/30;
    }
}
unit 40 {
    vlan-id 40;
    family inet {
        address 10.0.2.6/30;
    }
}
unit 60 {
    vlan-id 60;
    family inet {
        address 10.0.4.2/30;
    }
}
unit 200 {
    vlan-id 200;
    family inet {
        address 10.0.5.2/24;
    }
}
unit 210 {
    vlan-id 210;
    family inet {
        address 172.16.0.13/30;
    }
}
unit 230 {
    vlan-id 230;
    family inet {
        address 172.16.0.5/30;
    }
}
unit 270 {
    vlan-id 270;
    family inet {
        address 172.16.0.25/30;
    }
}
unit 280 {
    vlan-id 280;
    family inet {
        address 172.16.0.17/30;
    }
}
unit 300 {
    vlan-id 300;
```

Copyright © 2013, Juniper Networks, Inc.

```
            family inet {
              address 192.168.255.20/24;
            }
          }
        }
        lo0 {
          unit 0 {
            family inet {
              address 10.0.6.2/32;
            }
          }
          unit 1 {
            family inet {
              address 1.1.1.1/32;
            }
          }
          unit 2 {
            family inet {
              address 1.1.1.2/32;
            }
          }
          unit 3 {
            family inet {
              address 1.1.1.3/32;
            }
          }
        }
      }
      routing-options {
        static {
          route 192.168.200.0/24 next-hop 192.168.255.254;
        }
        rib-groups {
          static-cust-inet {
            import-rib [ static-cust-inet.inet.0 internet.inet.0 ];
            import-policy non-bgp-cust-ribgroup-policy;
          }
          static-cust-inet-interfaces {
            import-rib [ static-cust-inet.inet.0 internet.inet.0 ];
          }
          static-cust-i2 {
            import-rib [ static-cust-i2.inet.0 i2.inet.0 ];
            import-policy non-bgp-cust-ribgroup-policy;
          }
          static-cust-i2-interfaces {
            import-rib [ static-cust-i2.inet.0 i2.inet.0 ];
          }
          static-cust-all-feeds {
            import-rib [ static-cust-all-feeds.inet.0 internet.inet.0
            i2.inet.0 nlr.inet.0 ];
            import-policy static-all-feeds-cust-ribgroup-policy;
          }
          static-cust-all-feeds-interfaces {
            import-rib [ static-cust-all-feeds.inet.0 internet.inet.0
            i2.inet.0 nlr.inet.0 ];
          }
```


Copyright © 2013, Juniper Networks, Inc.

```
        bgp-customers-rg {
            import-rib [ bgp-customer.inet.0 internet.inet.0 i2.inet.0
            nlr.inet.0 ];
            import-policy static-all-feeds-cust-ribgroup-policy;
        }
        bgp-customer-rg-interfaces {
            import-rib [ bgp-customer.inet.0 internet.inet.0 i2.inet.0
            nlr.inet.0 ];
        }
        bgp-cust2-inet-interfaces {
            import-rib [ internet.inet.0 bgp-cust-2.inet.0 ];
        }
        bgp-cust2-i2-interfaces {
            import-rib [ i2.inet.0 bgp-cust-2.inet.0 ];
        }
        bgp-cust2-nlr-interfaces {
            import-rib [ nlr.inet.0 bgp-cust-2.inet.0 ];
        }
        bgp-cust2-interfaces {
            import-rib [ bgp-cust-2.inet.0 internet.inet.0 i2.inet.0
            nlr.inet.0 ];
        }
    }
    router-id 10.0.6.2;
    autonomous-system 65000;
}
policy-options {
    policy-statement bgp-cust-instance-import {
        term nlr {
            from {
                instance nlr;
                community nlr-vr;
            }
            then {
                local-preference 105;
                accept;
            }
        }
        term internet {
            from {
                instance internet;
                community internet-vr;
            }
            then accept;
        }
        term i2 {
            from {
                instance i2;
                community i2-vr;
            }
            then {
                local-preference 110;
                accept;
            }
        }
        term reject-all {
```

Copyright © 2013, Juniper Networks, Inc.

```
      then reject;
    }
  }
  policy-statement cust-d-export {
    term adv-providers {
      from community [ internet-vr i2-vr nlr-vr ];
      then accept;
    }
  }
  policy-statement cust-routes-import {
    term add-comm {
      then {
        community add cust-routes;
      }
    }
  }
  policy-statement i2-export-policy {
    term no-leaking {
      from {
        route-filter 0.0.0.0/0 through 0.0.0.0/32 reject;
      }
    }
    term adv-cust-routes {
      from community cust-routes;
      then accept;
    }
    term no-transit {
      from as-path null;
      then accept;
    }
    term reject-all {
      then reject;
    }
  }
  policy-statement i2-import-policy {
    term tag-i2-vr-community {
      then {
        community add i2-vr;
      }
    }
  }
  policy-statement i2-nlr-bgp-instance-import {
    term accept-i2 {
      from {
        instance i2;
        community i2-vr;
      }
      then {
        local-preference 110;
        accept;
      }
    }
    term accept-nlr {
      from {
        instance nlr;
        community nlr-vr;
```


Copyright © 2013, Juniper Networks, Inc.

```
        }
        then accept;
    }
    term reject-all {
        then reject;
    }
}
policy-statement internet-export-policy {
    term no-leaking {
        from {
            route-filter 0.0.0.0/0 through 0.0.0.0/32 reject;
        }
    }
    term prepend {
        from community prepend-twice;
        then as-path-prepend "65000 65000";
    }
    term adv-cust-routes {
        from community cust-routes;
        then accept;
    }
    term no-transit {
        from as-path null;
        then accept;
    }
    term reject-all {
        then reject;
    }
}
policy-statement internet-import-policy {
    term tag-vr-community {
        then {
            community add internet-vr;
        }
    }
}
policy-statement nlr-export-policy {
    term no-leaking {
        from {
            route-filter 0.0.0.0/0 through 0.0.0.0/32 reject;
        }
    }
    term adv-cust-routes {
        from community cust-routes;
        then accept;
    }
    term no-transit {
        from as-path null;
        then accept;
    }
    term reject-all {
        then reject;
    }
}
policy-statement nlr-import-policy {
    term tag-nlr-vr-community {
```

Copyright © 2013, Juniper Networks, Inc.

```
            then {
                community add nlr-vr;
            }
        }
    }
    policy-statement non-bgp-cust-ribgroup-policy {
        term reject-default {
            from {
                route-filter 0.0.0.0/0 exact reject;
            }
        }
        term cust-statics {
            from protocol static;
            then {
                community add cust-routes;
                accept;
            }
        }
        term reject-all {
            then reject;
        }
    }
    policy-statement static-all-feeds-cust-ribgroup-policy {
        term reject-default {
            from {
                route-filter 0.0.0.0/0 exact reject;
            }
        }
        term cust-statics {
            from protocol static;
            then {
                community add cust-routes;
                community add prepend-twice;
                accept;
            }
        }
        term cust-bgp {
            from protocol bgp;
            then {
                community add cust-routes;
                community add prepend-twice;
                accept;
            }
        }
        term reject-all {
            then reject;
        }
    }
    community cust-routes members 65000:1000;
    community i2-vr members 65000:222;
    community internet-vr members 65000:111;
    community nlr-vr members 65000:333;
    community prepend-twice members 65000:500;
    as-path null "()";
}
routing-instances {
```


Copyright © 2013, Juniper Networks, Inc.

```
bgp-cust-2 {
  instance-type virtual-router;
  interface ge-0/0/0.40;
  routing-options {
    interface-routes {
      rib-group inet bgp-cust2-interfaces;
    }
  }
}
bgp-customer {
  instance-type virtual-router;
  interface ge-0/0/0.30;
  routing-options {
    interface-routes {
      rib-group inet bgp-customer-rg-interfaces;
    }
    static {
      rib-group bgp-customers-rg;
    }
    instance-import bgp-cust-instance-import;
  }
  protocols {
    bgp {
      family inet {
        unicast {
          rib-group bgp-customers-rg;
        }
      }
      group customers {
        type external;
        advertise-inactive;
        neighbor 10.0.4.5 {
          export cust-d-export;
          peer-as 65100;
        }
      }
    }
  }
}
i2 {
  instance-type virtual-router;
  interface ge-0/0/0.210;
  interface ge-0/0/0.280;
  interface lo0.2;
  routing-options {
    interface-routes {
      rib-group inet bgp-cust2-i2-interfaces;
    }
  }
  protocols {
    bgp {
      group i2 {
        type external;
        traceoptions {
          file i2-bgp-trace;
          flag all detail;
```

Copyright © 2013, Juniper Networks, Inc.

```
            }
            advertise-inactive;
            import i2-import-policy;
            export i2-export-policy;
            peer-as 65222;
            multipath multiple-as;
            neighbor 172.16.0.14;
            neighbor 172.16.0.18;
          }
          group customers {
            type external;
            multihop {
              ttl 2;
            }
            local-address 1.1.1.2;
            advertise-inactive;
            import cust-routes-import;
            neighbor 10.0.2.5 {
              peer-as 65200;
            }
          }
        }
      }
    }
    internet {
      instance-type virtual-router;
      interface ge-0/0/0.230;
      interface ge-0/0/0.270;
      interface lo0.1;
      routing-options {
        interface-routes {
          rib-group inet bgp-cust2-inet-interfaces;
        }
      }
      protocols {
        bgp {
          group peer-1 {
            type external;
            advertise-inactive;
            import internet-import-policy;
            export internet-export-policy;
            peer-as 65010;
            multipath multiple-as;
            neighbor 172.16.0.6;
          }
          group peer-2 {
            type external;
            advertise-inactive;
            import internet-import-policy;
            export internet-export-policy;
            peer-as 65020;
            multipath multiple-as;
            neighbor 172.16.0.26;
          }
          group customers {
            type external;
```

 Copyright © 2013, Juniper Networks, Inc.

```
            traceoptions {
               file bgp-trace;
               flag open detail;
               flag state detail;
               flag general;
            }
            multihop {
               ttl 3;
            }
            local-address 1.1.1.1;
            advertise-inactive;
            import cust-routes-import;
            neighbor 10.0.2.5 {
               peer-as 65200;
            }
         }
      }
   }
}
nlr {
   instance-type virtual-router;
   interface ge-0/0/0.200;
   interface lo0.3;
   routing-options {
      interface-routes {
         rib-group inet bgp-cust2-nlr-interfaces;
      }
   }
   protocols {
      bgp {
         group nlr {
            type external;
            advertise-inactive;
            import nlr-import-policy;
            export nlr-export-policy;
            peer-as 65050;
            multipath multiple-as;
            neighbor 10.0.5.254;
         }
         group customers {
            type external;
            multihop {
               ttl 2;
            }
            local-address 1.1.1.3;
            advertise-inactive;
            import cust-routes-import;
            neighbor 10.0.2.5 {
               peer-as 65200;
            }
         }
      }
   }
}
static-cust-all-feeds {
   instance-type virtual-router;
```

Copyright © 2013, Juniper Networks, Inc.

```
        interface ge-0/0/0.10;
        routing-options {
          interface-routes {
            rib-group inet static-cust-all-feeds-interfaces;
          }
          static {
            rib-group static-cust-all-feeds;
            route 0.0.0.0/0 next-table internet.inet.0;
            route 192.168.50.0/24 next-hop 10.0.4.13;
          }
          instance-import i2-nlr-bgp-instance-import;
        }
      }
      static-cust-i2 {
        instance-type virtual-router;
        interface ge-0/0/0.20;
        routing-options {
          interface-routes {
            rib-group inet static-cust-i2-interfaces;
          }
          static {
            rib-group static-cust-i2;
            route 0.0.0.0/0 next-table i2.inet.0;
            route 192.168.40.0/24 next-hop 10.0.4.9;
          }
        }
      }
      static-cust-inet {
        instance-type virtual-router;
        interface ge-0/0/0.60;
        routing-options {
          interface-routes {
            rib-group inet static-cust-inet-interfaces;
          }
          static {
            rib-group static-cust-inet;
            route 0.0.0.0/0 next-table internet.inet.0;
            route 192.168.30.0/24 next-hop 10.0.4.1;
          }
        }
      }
    }
```

## Customer Device Configuration

This example uses the SRX210 Services Gateway as the customer device.

Use the **show** command to display the configuration of this device.

```
user@SRX210-A-R3> show | no-more

## Last changed: 2010-04-05 21:29:15 UTC
version 10.0R2.10;
system {
  host-name SRX210-A-R3;
  root-authentication {
    encrypted-password "$1"; ## SECRET-DATA
```


Copyright © 2013, Juniper Networks, Inc.

```
    }
    name-server {
        68.87.68.162;
        68.87.74.162;
    }
    login {
        user user {
            uid 2000;
            class super-user;
            authentication {
                encrypted-password "$1"; ## SECRET-DATA
            }
        }
    }
    services {
        ftp;
        ssh;
        telnet;
        web-management {
            http {
                interface fe-0/0/0.0;
            }
        }
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    fe-0/0/0 {
        vlan-tagging;
        unit 10 {
            vlan-id 10;
            family inet {
                address 10.0.4.13/30;
            }
        }
        unit 20 {
            vlan-id 20;
            family inet {
```

Copyright © 2013, Juniper Networks, Inc.

```
            address 10.0.4.9/30;
          }
        }
        unit 30 {
          vlan-id 30;
          family inet {
            address 10.0.4.5/30;
          }
        }
        unit 40 {
          vlan-id 40;
          family inet {
            address 10.0.2.5/30;
          }
        }
        unit 60 {
          vlan-id 60;
          family inet {
            address 10.0.4.1/30;
          }
        }
        unit 110 {
          vlan-id 110;
          family inet {
            address 10.0.2.2/30;
          }
        }
        unit 210 {
          disable;
          vlan-id 210;
          family inet {
            address 172.16.0.13/30;
          }
        }
        unit 280 {
          disable;
          vlan-id 280;
          family inet {
            address 172.16.0.17/30;
          }
        }
        unit 300 {
          vlan-id 300;
          family inet {
            address 192.168.255.30/24;
          }
        }
      }
      lo0 {
        unit 0 {
          family inet {
            address 10.0.3.3/32;
            address 192.168.30.1/32;
          }
          family iso {
            address 49.0002.0100.0000.3003.00;
```


Copyright © 2013, Juniper Networks, Inc.

```
        }
      }
      unit 10 {
        family inet {
          address 192.168.50.1/32;
        }
      }
      unit 20 {
        family inet {
          address 192.168.40.1/32;
        }
      }
      unit 30 {
        family inet {
          address 192.168.60.1/32;
        }
      }
    }
  }
  routing-options {
    static {
      route 192.168.200.0/24 {
        next-hop 192.168.255.254;
        no-readvertise;
      }
      route 192.168.30.0/24 discard;
      route 0.0.0.0/0 next-hop 10.0.4.2;
    }
    router-id 10.0.3.3;
    autonomous-system 65000;
  }
  policy-options {
    policy-statement adv-static-to-bgp {
      term 1 {
        from {
          protocol static;
          route-filter 192.168.0.0/16 prefix-length-range /24-/24;
        }
        then accept;
      }
    }
    policy-statement cust-d-export {
      term 1 {
        from {
          route-filter 192.168.60.0/24 exact accept;
        }
      }
    }
    policy-statement cust-e-export {
      term 1 {
        from {
          route-filter 192.168.90.0/24 exact accept;
        }
      }
    }
    policy-statement isis-summ {
```

```
    term 3 {
        from {
            route-filter 10.0.5.0/24 exact;
        }
        to level 2;
        then accept;
    }
    term 1 {
        from {
            protocol aggregate;
            route-filter 10.0.4.0/22 exact;
        }
        to level 2;
        then accept;
    }
    term 2 {
        from {
            route-filter 10.0.4.0/22 longer;
        }
        to level 2;
        then reject;
    }
    term 4 {
        from {
            level 2;
            route-filter 10.0.3.0/29 longer;
        }
        to level 1;
        then accept;
    }
}
policy-statement nhs {
    term 1 {
        from community t-routes;
        then {
            next-hop self;
        }
    }
}
policy-statement t-export {
    term send-agg {
        from {
            route-filter 10.0.0.0/8 longer reject;
            route-filter 172.16.40.0/29 longer reject;
            route-filter 192.168.0.0/22 longer reject;
        }
        then as-path-prepend "65412 65412";
    }
}
policy-statement transit-import {
    term rfc1918 {
        from {
            route-filter 10.0.0.0/8 orlonger reject;
            route-filter 172.16.0.0/12 orlonger reject;
            route-filter 192.168.0.0/16 orlonger reject;
            route-filter 0.0.0.0/0 through 0.0.0.0/32 reject;
```


Copyright © 2013, Juniper Networks, Inc.

```
        }
      }
      term no-27+ {
        from {
          route-filter 0.0.0.0/0 prefix-length-range /27-/32 reject;
        }
      }
      term transit-comm {
        then {
          community add t-routes;
        }
      }
    }
    community all members *:*;
    community c1-routes members 65412:65010;
    community c2-routes members 65412:65020;
    community p1-routes members 65412:65050;
    community t-routes members 65412:65222;
}
security {
    inactive: screen {
      ids-option untrust-screen {
        icmp {
          ping-death;
        }
        ip {
          source-route-option;
          tear-drop;
        }
        tcp {
          syn-flood {
            alarm-threshold 1024;
            attack-threshold 200;
            source-threshold 1024;
            destination-threshold 2048;
            queue-size 2000;
            timeout 20;
          }
          land;
        }
      }
    }
    zones {
      security-zone trust {
        tcp-rst;
        interfaces {
          all {
            host-inbound-traffic {
              system-services {
                all;
              }
              protocols {
                all;
              }
            }
          }
```

Copyright © 2013, Juniper Networks, Inc.

```
            }
          }
          security-zone untrust;
        }
        forwarding-options {
          family {
            mpls {
              mode packet-based;
            }
          }
        }
      }
      routing-instances {
        cust-b {
          instance-type virtual-router;
          interface fe-0/0/0.20;
          interface lo0.20;
          routing-options {
            static {
              route 0.0.0.0/0 next-hop 10.0.4.10;
            }
          }
        }
        cust-c {
          instance-type virtual-router;
          interface fe-0/0/0.10;
          interface lo0.10;
          routing-options {
            static {
              route 0.0.0.0/0 next-hop 10.0.4.14;
            }
          }
        }
        cust-d {
          instance-type virtual-router;
          interface fe-0/0/0.30;
          interface lo0.30;
          routing-options {
            static {
              route 192.168.60.0/24 discard;
            }
            autonomous-system 65100;
          }
          protocols {
            bgp {
              group provider-ebgp {
                type external;
                export cust-d-export;
                peer-as 65000;
                neighbor 10.0.4.6;
              }
            }
          }
        }
        cust-e {
          instance-type virtual-router;
```


Copyright © 2013, Juniper Networks, Inc.

```
    interface fe-0/0/0.40;
    routing-options {
      static {
        route 1.1.1.0/29 next-hop 10.0.2.6;
        route 192.168.90.0/24 discard;
      }
      autonomous-system 65200;
    }
    protocols {
      bgp {
        traceoptions {
          file cust-e-trace;
          flag open detail;
          flag keepalive;
          flag state detail;
        }
        group provider-ebgp {
          type external;
          multihop {
            ttl 3;
          }
          export cust-e-export;
          peer-as 65000;
          neighbor 1.1.1.1;
          neighbor 1.1.1.2;
          neighbor 1.1.1.3;
        }
      }
    }
  }
}
```

## Service Provider Device Configuration

This example uses the EX3200 Ethernet Switch as the service provider device.

Use the **show configuration** command to display the configuration of this device.

```
user@EX-3200-1> show configuration | no-more

  system {
    host-name EX-3200-1;
    time-zone America/New_York;
    root-authentication {
      encrypted-password "$1"; ## SECRET-DATA
    }
    name-server {
      68.87.68.162;
      68.87.74.162;
    }
    login {
      user user {
        uid 2000;
        class super-user;
        authentication {
          encrypted-password "$1"; ## SECRET-DATA
        }
```

```
        }
        user trusteduser {
            uid 2002;
            class super-user;
            authentication {
                encrypted-password "$1"; ## SECRET-DATA
            }
        }
    }
    services {
        ftp;
        ssh;
        telnet;
        web-management {
            http;
        }
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any notice;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
}
    chassis {
        aggregated-devices {
            ethernet {
                device-count 5;
            }
        }
    }
    interfaces {
        ge-0/0/1 {
            description J2350-1;
            disable;
            unit 0 {
                family ethernet-switching {
                    port-mode trunk;
                    vlan {
                        members [ 10 200 30 50 300 ];
                    }
                }
            }
        }
        ge-0/0/2 {
            description J2350-2;
            unit 0 {
                family ethernet-switching {
                    port-mode trunk;
                    vlan {
```

 Copyright © 2013, Juniper Networks, Inc.

```
            members [ 20 30 60 200 300 230 210 280 270 10 40 ];
          }
        }
      }
    }
    ge-0/0/3 {
      description SRX210-A;
      unit 0 {
        family ethernet-switching {
          port-mode trunk;
          vlan {
            members [ 10 210 40 60 70 110 300 280 30 20 ];
          }
        }
      }
    }
    ge-0/0/4 {
      description SRX210-B;
      disable;
      unit 0 {
        family ethernet-switching {
          port-mode trunk;
          vlan {
            members [ 20 50 40 130 80 300 230 240 ];
          }
        }
      }
    }
    ge-0/0/5 {
      description J4300;
      disable;
      unit 0 {
        family ethernet-switching {
          port-mode trunk;
          vlan {
            members [ 110 130 120 140 300 ];
          }
        }
      }
    }
    ge-0/0/6 {
      description SRX100-1;
      disable;
      unit 0 {
        family ethernet-switching {
          port-mode trunk;
          vlan {
            members [ 220 70 120 90 250 300 ];
          }
        }
      }
    }
    ge-0/0/7 {
      description SRX100-2;
      disable;
      unit 0 {
```

Copyright © 2013, Juniper Networks, Inc.

```
        family ethernet-switching {
          port-mode trunk;
          vlan {
            members [ 240 80 140 90 260 300 270 ];
          }
        }
      }
    }
    ge-0/0/14 {
      vlan-tagging;
      unit 800 {
        vlan-id 800;
        family inet {
          address 8.8.8.1/30;
        }
      }
      unit 900 {
        vlan-id 900;
        family inet {
          address 9.9.9.1/30;
        }
      }
    }
    ge-0/0/15 {
      vlan-tagging;
      unit 801 {
        vlan-id 800;
        family inet {
          address 8.8.8.2/30;
        }
      }
      unit 901 {
        vlan-id 900;
        family inet {
          address 9.9.9.2/30;
        }
      }
    }
    ge-0/0/23 {
      description J2300-headend;
      unit 0 {
        family ethernet-switching {
          port-mode trunk;
          vlan {
            members 300;
          }
        }
      }
    }
    lo0 {
      unit 1 {
        family inet {
          address 192.168.0.1/32;
          address 192.168.1.1/32;
          address 192.168.2.1/32;
          address 192.168.3.1/32;
```


Copyright © 2013, Juniper Networks, Inc.

```
            address 192.168.4.1/32;
          }
        }
        unit 10 {
          family inet {
            address 200.200.1.1/32;
          }
        }
        unit 11 {
          family inet {
            address 201.201.1.1/32;
          }
        }
        unit 500 {
          family inet {
            address 120.120.1.1/32;
          }
        }
        unit 800 {
          family inet {
            address 130.130.1.1/32;
          }
        }
        unit 801 {
          family inet {
            address 130.130.2.2/32;
          }
        }
      }
      vlan {
        unit 200 {
          family inet {
            address 10.0.5.254/24;
          }
        }
        unit 210 {
          family inet {
            address 172.16.0.14/30;
          }
        }
        unit 220 {
          family inet {
            address 172.16.0.22/30;
          }
        }
        unit 230 {
          family inet {
            address 172.16.0.6/30;
          }
        }
        unit 240 {
          family inet {
            address 172.16.0.10/30;
          }
        }
        unit 250 {
```

```
        family inet {
          address 172.16.40.1/30;
        }
      }
      unit 260 {
        family inet {
          address 172.16.40.5/30;
        }
      }
      unit 270 {
        family inet {
          address 172.16.0.26/30;
        }
      }
      unit 280 {
        family inet {
          address 172.16.0.18/30;
        }
      }
      unit 300 {
        family inet {
          address 192.168.255.100/24;
        }
      }
    }
  }
  routing-options {
    static {
      route 0.0.0.0/0 {
        next-hop 192.168.255.254;
        no-readvertise;
      }
    }
    autonomous-system 1;
  }
  protocols {
    rstp;
  }
  policy-options {
    policy-statement C1-export {
      term 1 {
        from {
          protocol static;
          route-filter 200.200.0.0/16 exact accept;
          route-filter 10.0.0.0/8 longer accept;
        }
      }
      term 2 {
        from protocol bgp;
        then accept;
      }
    }
    policy-statement C2-export {
      term 1 {
        from {
          protocol static;
```


Copyright © 2013, Juniper Networks, Inc.

```
            route-filter 201.201.0.0/16 exact accept;
            route-filter 10.0.0.0/8 longer accept;
          }
        }
      }
      policy-statement P1-export {
        term 1 {
          from {
            protocol static;
            route-filter 120.120.0.0/16 exact accept;
            route-filter 10.0.0.0/8 longer accept;
          }
        }
      }
      policy-statement TX-export {
        term 1 {
          from {
            protocol static;
            route-filter 130.130.0.0/16 exact accept;
            route-filter 10.0.0.0/8 longer accept;
          }
        }
      }
      policy-statement ospf-out {
        term 1 {
          from {
            route-filter 192.168.0.0/22 orlonger;
          }
          then accept;
        }
        term default {
          then reject;
        }
      }
      policy-statement rip-routes {
        term 1 {
          from protocol [ static rip ];
          then accept;
        }
      }
    }
    routing-instances {
      C1 {
        instance-type virtual-router;
        interface ge-0/0/14.800;
        interface lo0.10;
        interface vlan.230;
        routing-options {
          static {
            route 200.200.0.0/16 discard;
            route 10.0.3.4/32 next-hop [ 172.16.0.5 172.16.0.9 ];
            route 10.10.1.0/24 discard;
            route 10.10.2.0/24 discard;
            route 10.10.3.0/24 discard;
            route 10.10.4.0/24 discard;
            route 10.10.5.0/24 discard;
```

Copyright © 2013, Juniper Networks, Inc.

```
        route 10.10.6.0/24 discard;
        route 10.10.7.0/24 discard;
        route 10.10.8.0/24 discard;
        route 10.10.9.0/24 discard;
        route 10.10.10.0/24 discard;
        route 10.10.11.0/24 discard;
        route 10.10.12.0/24 discard;
        route 10.10.13.0/24 discard;
        route 10.10.14.0/24 discard;
        route 10.10.15.0/24 discard;
        route 10.10.16.0/24 discard;
        route 10.10.17.0/24 discard;
        route 10.10.18.0/24 discard;
        route 10.10.19.0/24 discard;
        route 10.10.20.0/24 discard;
      }
      autonomous-system 65010;
    }
    protocols {
      bgp {
        group as65010 {
          type external;
          advertise-inactive;
          export C1-export;
          peer-as 65000;
          local-as 65010;
          neighbor 8.8.8.2 {
            peer-as 65020;
          }
          neighbor 172.16.0.5;
        }
      }
    }
  }
  C2 {
    instance-type virtual-router;
    interface ge-0/0/15.801;
    interface lo0.11;
    interface vlan.270;
    routing-options {
      static {
        route 201.201.0.0/16 discard;
        route 0.0.0.0/0 discard;
        route 201.201.0.64/26 discard;
        route 201.201.1.0/24 discard;
        route 201.201.2.0/24 discard;
        route 201.201.3.0/24 discard;
        route 201.201.4.0/24 discard;
        route 201.201.5.0/24 discard;
        route 201.201.6.0/24 discard;
        route 201.201.7.0/24 discard;
        route 10.10.1.0/24 discard;
        route 10.10.2.0/24 discard;
        route 10.10.3.0/24 discard;
        route 10.10.4.0/24 discard;
        route 10.10.5.0/24 discard;
```


Copyright © 2013, Juniper Networks, Inc.

```
        route 10.10.6.0/24 discard;
        route 10.10.7.0/24 discard;
        route 10.10.8.0/24 discard;
        route 10.10.9.0/24 discard;
        route 10.10.10.0/24 discard;
        route 10.10.11.0/24 discard;
        route 10.10.12.0/24 discard;
        route 10.10.13.0/24 discard;
        route 10.10.14.0/24 discard;
        route 10.10.15.0/24 discard;
        route 10.10.16.0/24 discard;
        route 10.10.17.0/24 discard;
        route 10.10.18.0/24 discard;
        route 10.10.19.0/24 discard;
        route 10.10.20.0/24 discard;
      }
      autonomous-system 65020;
    }
    protocols {
      bgp {
        group as65020 {
          type external;
          advertise-inactive;
          export C2-export;
          local-as 65020;
          neighbor 172.16.0.25 {
            peer-as 65000;
          }
          neighbor 8.8.8.1 {
            peer-as 65010;
          }
        }
      }
    }
  }
  DC1 {
    instance-type virtual-router;
    interface lo0.1;
    interface vlan.250;
    interface vlan.260;
    routing-options {
      static {
        route 192.168.0.0/24 receive;
        route 192.168.1.0/24 receive;
        route 192.168.2.0/24 receive;
        route 192.168.3.0/24 receive;
      }
    }
    protocols {
      ospf {
        export ospf-out;
        area 0.0.0.0 {
          interface vlan.250;
          interface vlan.260;
        }
      }
```

Copyright © 2013, Juniper Networks, Inc.

```
    inactive: rip {
      group rip {
        export rip-routes;
        neighbor vlan.250;
        neighbor vlan.260;
      }
    }
  }
}
P1 {
  instance-type virtual-router;
  interface lo0.500;
  interface vlan.200;
  routing-options {
    static {
      route 120.120.0.0/16 discard;
      route 10.10.11.0/24 discard;
      route 10.10.12.0/24 discard;
      route 10.10.13.0/24 discard;
      route 10.10.14.0/24 discard;
    }
    autonomous-system 65050;
  }
  protocols {
    bgp {
      group as65050 {
        type external;
        advertise-inactive;
        export P1-export;
        peer-as 65000;
        local-as 65050;
        neighbor 10.0.5.1;
        neighbor 10.0.5.2;
      }
    }
  }
}
T1 {
  instance-type virtual-router;
  interface ge-0/0/14.900;
  interface lo0.800;
  interface vlan.210;
  routing-options {
    static {
      route 130.130.0.0/16 discard;
      route 130.130.2.2/32 next-hop 9.9.9.2;
      route 10.10.1.0/24 discard;
      route 10.10.2.0/24 discard;
      route 10.10.3.0/24 discard;
      route 10.10.4.0/24 discard;
      route 10.10.5.0/24 discard;
    }
    autonomous-system 65222;
  }
  protocols {
    bgp {
```


Copyright © 2013, Juniper Networks, Inc.

```
            export TX-export;
            group as65222 {
              type external;
              advertise-inactive;
              peer-as 65000;
              local-as 65222;
              neighbor 172.16.0.13;
            }
            group ibgp {
              type internal;
              local-address 130.130.1.1;
              local-as 65222;
              neighbor 130.130.2.2;
            }
          }
        }
      }
      T2 {
        instance-type virtual-router;
        interface ge-0/0/15.901;
        interface lo0.801;
        interface vlan.220;
        interface vlan.280;
        routing-options {
          static {
            route 130.130.0.0/16 discard;
            route 130.130.1.1/32 next-hop 9.9.9.1;
            route 10.10.1.0/24 discard;
            route 10.10.2.0/24 discard;
            route 10.10.3.0/24 discard;
            route 10.10.4.0/24 discard;
            route 10.10.5.0/24 discard;
          }
          autonomous-system 65222;
        }
        protocols {
          bgp {
            export TX-export;
            group ibgp {
              type internal;
              local-address 130.130.2.2;
              local-as 65222;
              neighbor 130.130.1.1;
            }
            group as65222 {
              type external;
              advertise-inactive;
              peer-as 65000;
              local-as 65222;
              neighbor 172.16.0.17;
              neighbor 172.16.0.21;
            }
          }
        }
      }
    }
```

Copyright © 2013, Juniper Networks, Inc.

```
vlans {
    vlan10 {
        vlan-id 10;
    }
    vlan110 {
        vlan-id 110;
    }
    vlan120 {
        vlan-id 120;
    }
    vlan130 {
        vlan-id 130;
    }
    vlan140 {
        vlan-id 140;
    }
    vlan20 {
        vlan-id 20;
    }
    vlan200 {
        vlan-id 200;
        l3-interface vlan.200;
    }
    vlan210 {
        vlan-id 210;
        l3-interface vlan.210;
    }
    vlan220 {
        vlan-id 220;
        l3-interface vlan.220;
    }
    vlan230 {
        vlan-id 230;
        l3-interface vlan.230;
    }
    vlan240 {
        vlan-id 240;
        l3-interface vlan.240;
    }
    vlan250 {
        vlan-id 250;
        l3-interface vlan.250;
    }
    vlan260 {
        vlan-id 260;
        l3-interface vlan.260;
    }
    vlan270 {
        vlan-id 270;
        l3-interface vlan.270;
    }
    vlan280 {
        vlan-id 280;
        l3-interface vlan.280;
    }
    vlan30 {
```


Copyright © 2013, Juniper Networks, Inc.

```
        vlan-id 30;
      }
      vlan300 {
        vlan-id 300;
        l3-interface vlan.300;
      }
      vlan40 {
        vlan-id 40;
      }
      vlan50 {
        vlan-id 50;
      }
      vlan60 {
        vlan-id 60;
      }
      vlan70 {
        vlan-id 70;
      }
      vlan80 {
        vlan-id 80;
      }
      vlan90 {
        vlan-id 90;
      }
    }
```

## Complete Routing Table

Following is the complete routing table view of the Educational Network (New University):

```
user@J2350-2-R2# run show route
 i2.inet.0: 23 destinations, 29 routes (23 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
1.1.1.2/32 *[Direct/0] 04:43:57
> via lo0.2
10.0.2.4/30 *[Direct/0] 03:38:59
> via ge-0/0/0.40
10.0.2.6/32 *[Local/0] 03:32:36
Local via ge-0/0/0.40
10.0.4.4/30 *[Direct/0] 03:38:59
> via ge-0/0/0.30
10.0.4.6/32 *[Local/0] 03:32:36
Local via ge-0/0/0.30
10.0.4.8/30 *[Direct/0] 04:31:48
> via ge-0/0/0.20
10.0.4.10/32 *[Local/0] 03:32:36
Local via ge-0/0/0.20
10.0.4.12/30 *[Direct/0] 03:38:59
> via ge-0/0/0.10
10.0.4.14/32 *[Local/0] 03:32:36
Local via ge-0/0/0.10
10.10.1.0/24 *[BGP/170] 2d 23:53:44, localpref 100, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
[BGP/170] 2d 23:40:26, localpref 100
AS path: 65222 I
> to 172.16.0.18 via ge-0/0/0.280
10.10.2.0/24 *[BGP/170] 2d 23:53:44, localpref 100, from 172.16.0.14
```

Copyright © 2013, Juniper Networks, Inc.

```
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
[BGP/170] 2d 23:40:26, localpref 100
AS path: 65222 I
> to 172.16.0.18 via ge-0/0/0.280
10.10.3.0/24 *[BGP/170] 2d 23:53:44, localpref 100, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
[BGP/170] 2d 23:40:26, localpref 100
AS path: 65222 I
> to 172.16.0.18 via ge-0/0/0.280
10.10.4.0/24 *[BGP/170] 2d 23:53:44, localpref 100, from 172.16.0.14
AS path: 65222 I
to 172.16.0.14 via ge-0/0/0.210
> to 172.16.0.18 via ge-0/0/0.280
[BGP/170] 2d 23:40:26, localpref 100
AS path: 65222 I
> to 172.16.0.18 via ge-0/0/0.280
10.10.5.0/24 *[BGP/170] 2d 23:53:44, localpref 100
AS path: 65222 I
> to 172.16.0.14 via ge-0/0/0.210
to 172.16.0.18 via ge-0/0/0.280
[BGP/170] 2d 23:40:26, localpref 100
AS path: 65222 I
> to 172.16.0.18 via ge-0/0/0.280
130.130.0.0/16 *[BGP/170] 2d 23:53:44, localpref 100
AS path: 65222 I
> to 172.16.0.14 via ge-0/0/0.210
to 172.16.0.18 via ge-0/0/0.280
[BGP/170] 2d 23:40:26, localpref 100
AS path: 65222 I
> to 172.16.0.18 via ge-0/0/0.280
172.16.0.12/30 *[Direct/0] 3d 03:19:15
> via ge-0/0/0.210
172.16.0.13/32 *[Local/0] 3d 03:19:15
Local via ge-0/0/0.210
172.16.0.16/30 *[Direct/0] 3d 03:19:15
> via ge-0/0/0.280
172.16.0.17/32 *[Local/0] 3d 03:19:15
Local via ge-0/0/0.280
192.168.40.0/24 *[Static/5] 03:32:36
> to 10.0.4.9 via ge-0/0/0.20
192.168.50.0/24 *[Static/5] 03:32:36
> to 10.0.4.13 via ge-0/0/0.10
192.168.60.0/24 *[BGP/170] 03:32:36, localpref 100
AS path: 65100 I
> to 10.0.4.5 via ge-0/0/0.30
192.168.90.0/24 *[BGP/170] 03:33:42, localpref 100
AS path: 65200 I
> to 10.0.2.5 via ge-0/0/0.40
internet.inet.0: 40 destinations, 63 routes (40 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
1.1.1.1/32 *[Direct/0] 04:43:57
> via lo0.1
10.0.2.4/30 *[Direct/0] 03:38:59
> via ge-0/0/0.40
10.0.2.6/32 *[Local/0] 03:32:36
Local via ge-0/0/0.40
10.0.3.4/32 *[BGP/170] 3d 02:58:01, localpref 100
```


Copyright © 2013, Juniper Networks, Inc.

```
AS path: 65020 65010 I
> to 172.16.0.26 via ge-0/0/0.270
10.0.4.0/30 *[Direct/0] 04:31:48
> via ge-0/0/0.60
10.0.4.2/32 *[Local/0] 03:32:36
Local via ge-0/0/0.60
10.0.4.4/30 *[Direct/0] 03:38:59
> via ge-0/0/0.30
10.0.4.6/32 *[Local/0] 03:32:36
Local via ge-0/0/0.30
10.0.4.12/30 *[Direct/0] 03:38:59
> via ge-0/0/0.10
10.0.4.14/32 *[Local/0] 03:32:36
Local via ge-0/0/0.10
10.10.1.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.2.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.3.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.4.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.5.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.6.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.7.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
```

Copyright © 2013, Juniper Networks, Inc.

```
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.8.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.9.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.10.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.11.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.12.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.13.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.14.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.15.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.16.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
```


Copyright © 2013, Juniper Networks, Inc.

```
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.17.0/24 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.18.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.19.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
10.10.20.0/24 *[BGP/170] 3d 02:58:05, localpref 100, from 172.16.0.6
AS path: 65010 I
to 172.16.0.6 via ge-0/0/0.230
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
172.16.0.4/30 *[Direct/0] 3d 03:19:15
> via ge-0/0/0.230
172.16.0.5/32 *[Local/0] 3d 03:19:15
Local via ge-0/0/0.230
172.16.0.24/30 *[Direct/0] 3d 03:19:15
> via ge-0/0/0.270
172.16.0.25/32 *[Local/0] 3d 03:19:15
Local via ge-0/0/0.270
192.168.30.0/24 *[Static/5] 03:32:36
> to 10.0.4.1 via ge-0/0/0.60
192.168.50.0/24 *[Static/5] 03:32:36
> to 10.0.4.13 via ge-0/0/0.10
192.168.60.0/24 *[BGP/170] 03:32:36, localpref 100
AS path: 65100 I
> to 10.0.4.5 via ge-0/0/0.30
192.168.90.0/24 *[BGP/170] 03:33:46, localpref 100
AS path: 65200 I
> to 10.0.2.5 via ge-0/0/0.40
200.200.0.0/16 *[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 I
> to 172.16.0.6 via ge-0/0/0.230
[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 65010 I
> to 172.16.0.26 via ge-0/0/0.270
201.201.0.0/16 *[BGP/170] 3d 02:58:01, localpref 100
AS path: 65020 I
> to 172.16.0.26 via ge-0/0/0.270
[BGP/170] 3d 02:58:05, localpref 100
AS path: 65010 65020 I
```

Copyright © 2013, Juniper Networks, Inc.

```
> to 172.16.0.6 via ge-0/0/0.230
nlr.inet.0: 17 destinations, 17 routes (17 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
1.1.1.3/32 *[Direct/0] 03:38:59
> via lo0.3
10.0.2.4/30 *[Direct/0] 03:38:59
> via ge-0/0/0.40
10.0.2.6/32 *[Local/0] 03:32:36
Local via ge-0/0/0.40
10.0.4.4/30 *[Direct/0] 03:38:59
> via ge-0/0/0.30
10.0.4.6/32 *[Local/0] 03:32:36
Local via ge-0/0/0.30
10.0.4.12/30 *[Direct/0] 03:38:59
> via ge-0/0/0.10
10.0.4.14/32 *[Local/0] 03:32:36
Local via ge-0/0/0.10
10.0.5.0/24 *[Direct/0] 03:38:59
> via ge-0/0/0.200
10.0.5.2/32 *[Local/0] 03:38:59
Local via ge-0/0/0.200
10.10.11.0/24 *[BGP/170] 03:38:51, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
10.10.12.0/24 *[BGP/170] 03:38:51, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
10.10.13.0/24 *[BGP/170] 03:38:51, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
10.10.14.0/24 *[BGP/170] 03:38:51, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
120.120.0.0/16 *[BGP/170] 03:38:51, localpref 100
AS path: 65050 I
> to 10.0.5.254 via ge-0/0/0.200
192.168.50.0/24 *[Static/5] 03:32:36
> to 10.0.4.13 via ge-0/0/0.10
192.168.60.0/24 *[BGP/170] 03:32:36, localpref 100
AS path: 65100 I
> to 10.0.4.5 via ge-0/0/0.30
192.168.90.0/24 *[BGP/170] 03:33:38, localpref 100
AS path: 65200 I
> to 10.0.2.5 via ge-0/0/0.40
```

**Related Documentation**

- Using Virtual Routers to Provide Customer Peering on page 1
- Example: Configuring Virtual Routers for Educational Networks on page 12


Copyright © 2013, Juniper Networks, Inc.