

Network Configuration Example

Configuring the BGP Local Autonomous System Attribute

Release
13.1



Published: 2013-02-08

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 Configuring the BGP Local Autonomous System Attribute

Release 13.1

NCE0045

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

Introduction	1
Understanding the BGP Local AS Attribute	1
Example: Configuring a Local AS for EBGp Sessions	5
Example: Configuring a Private Local AS for EBGp Sessions	15

Introduction

This document provides an overview and describes how to configure the BGP local autonomous system (AS) attribute. The BGP local AS attribute allows a router to impersonate a second autonomous system, providing a method for merging networks in a transition period with minimal disruptions in service. This is useful in situations when Internet service providers (ISPs) are merging previously separate networks.

Understanding the BGP Local AS Attribute

When an Internet service provider (ISP) acquires a network that belongs to a different autonomous system (AS), there is no seamless method for moving the BGP peers of the acquired network to the AS of the acquiring ISP. The process of configuring the BGP peers with the new AS number can be time-consuming and cumbersome. Sometimes customers do not want to or are not immediately able to modify their peer arrangements or configuration. During this kind of transition period, it can be useful to configure BGP-enabled devices in the new AS to use the former AS number in BGP updates. This former AS number is called a *local AS*.

Using a local AS number permits the routing devices in an acquired network to appear to belong to the former AS.

For example, ISP A, with an AS of 200, acquires ISP B, with an AS of 250. ISP B has a customer, ISP C, that does not want to change its configuration. After ISP B becomes part of ISP A, a local AS number of 250 is configured for use in EBGP peer sessions with ISP C. Consequently, the local AS number of 250 is either prepended before or used instead of the global AS number of 200 in the AS path used to export routes to direct external peers in ISP C.

If the route is received from an internal BGP (IBGP) peer, the AS path includes the local AS number prepended before the global AS number.

The local AS number is used instead of the global AS number if the route is an external route, such as a static route or an interior gateway protocol (IGP) route that is imported into BGP. If the route is external and you want the global AS number to be included in the AS path, you can apply a routing policy that uses **as-path-expand** or **as-path-prepend**. Use the **as-path-expand** policy action to place the global AS number behind the local AS number. Use the **as-path-prepend** policy action to place the global AS number in front of the local AS number.

For example:

```
user@R2# show policy-options
policy-statement prepend-global {
  term 1 {
    from protocol static;
    then {
      as-path-prepend 200; # or use as-path-expand
      accept;
    }
  }
}
```

```
}
user@R2# show protocols bgp
group ext {
  export prepend-global;
  type external;
  local-as 250;
  neighbor 10.0.0.1 {
    peer-as 100;
  }
  neighbor 10.1.0.2 {
    peer-as 300;
  }
}

user@R2# show routing-options
static {
  route 1.1.1.1/32 next-hop 10.0.0.1;
}
autonomous-system 200;

user@R3# run show route 1.1.1.1 protocol bgp
inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1.1.1.1/32          *[BGP/170] 00:05:11, localpref 100
                   AS path: 200 250 I, validation-state: unverified
                   > to 10.1.0.1 via lt-1/2/0.4
```

In a Layer 3 VPN scenario, in which a provider edge (PE) device uses external BGP (EBGP) to peer with a customer edge (CE) device, the **local-as** statement behaves differently than in the non-VPN scenario. In the VPN scenario, the global AS number defined in the master instance is prepended to the AS path by default. To override this behavior, you can configure the **no-prepend-global-as** in the routing-instance BGP configuration on the PE device, as shown here:

```
user@R2# show routing-instances
red {
  instance-type vrf;
  interface fe-1/2/0.2;
  route-distinguisher 2:1;
  vrf-target target:2:1;
  protocols {
    bgp {
      group toR1 {
        type external;
        peer-as 1;
        local-as 200 no-prepend-global-as;
        neighbor 10.1.1.1;
      }
    }
  }
}
```

The Junos operating system (Junos OS) implementation of the local AS attribute supports the following options:

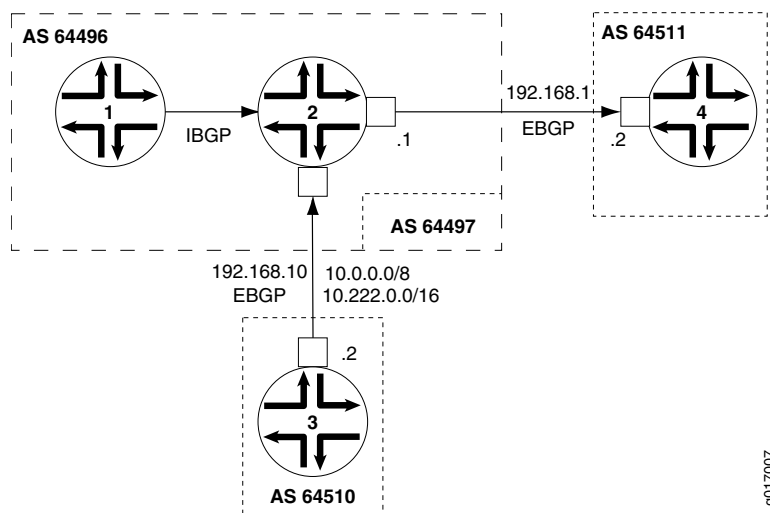
- **Local AS with private option**—When you use the **private** option, the local AS is used during the establishment of the BGP session with an EBGP neighbor but is hidden in the AS path sent to other EBGP peers. Only the global AS is included in the AS path sent to external peers.

The **private** option is useful for establishing local peering with routing devices that remain configured with their former AS or with a specific customer that has not yet modified its peer arrangements. The local AS is used to establish the BGP session with the EBGP neighbor but is hidden in the AS path sent to external peers in another AS.

Include the **private** option so that the local AS is not prepended before the global AS in the AS path sent to external peers. When you specify the **private** option, the local AS is prepended only in the AS path sent to the EBGP neighbor.

For example, in [Figure 1 on page 3](#), Router 1 and Router 2 are in AS 64496, Router 4 is in AS 64511, and Router 3 is in AS 64510. Router 2 formerly belonged to AS 64497, which has merged with another network and now belongs to AS 64496. Because Router 3 still peers with Router 2 using its former AS (64497), Router 2 needs to be configured with a local AS of 64497 in order to maintain peering with Router 3. Configuring a local AS of 64497 permits Router 2 to add AS 64497 when advertising routes to Router 3. Router 3 sees an AS path of 64497 64496 for the prefix 10/8.

Figure 1: Local AS Configuration



To prevent Router 2 from adding the local AS number in its announcements to other peers, use the **local-as 64497 private** statement. This statement configures Router 2 to not include local AS 64497 when announcing routes to Router 1 and to Router 4. In this case, Router 4 sees an AS path of 64496 64510 for the prefix 10.222/16.

- **Local AS with alias option**—In Junos OS Release 9.5 and later, you can configure a local AS as an alias. During the establishment of the BGP open session, the AS used in the open message alternates between the local AS and the global AS. If the local AS is used to connect with the EBGP neighbor, then only the local AS is prepended to the AS path when the BGP peer session is established. If the global AS is used to connect with the EBGP neighbor, then only the global AS is prepended to the AS path when the BGP peer session is established. The use of the **alias** option also means that

the local AS is not prepended to the AS path for any routes learned from that EBGp neighbor. Therefore, the local AS remains hidden from other external peers.

Configuring a local AS with the **alias** option is especially useful when you are migrating the routing devices in an acquired network to the new AS. During the migration process, some routing devices might be configured with the new AS while others remain configured with the former AS. For example, it is good practice to start by first migrating to the new AS any routing devices that function as route reflectors. However, as you migrate the route reflector clients incrementally, each route reflector has to peer with routing devices configured with the former AS, as well as peer with routing devices configured with the new AS. To establish local peer sessions, it can be useful for the BGP peers in the network to use both the local AS and the global AS. At the same time, you want to hide this local AS from external peers and use only the global AS in the AS path when exporting routes to another AS. In this kind of situation, configure the **alias** option.

Include the **alias** option to configure the local AS as an alias to the global AS configured at the **[edit routing-options]** hierarchy level. When you configure a local AS as an alias, during the establishment of the BGP open session, the AS used in the open message alternates between the local AS and the global AS. The local AS is prepended to the AS path only when the peer session with an EBGp neighbor is established using that local AS. The local AS is hidden in the AS path sent to any other external peers. Only the global AS is prepended to the AS path when the BGP session is established using the global AS.



NOTE: The **private** and **alias** options are mutually exclusive. You cannot configure both options with the same **local-as** statement.

- **Local AS with option not to prepend the global AS**—In Junos OS Release 9.6 and later, you can configure a local AS with the option not to prepend the global AS. Only the local AS is included in the AS path sent to external peers.

Use the **no-prepend-global-as** option when you want to strip the global AS number from outbound BGP updates in a virtual private network (VPN) scenario. This option is useful in a VPN scenario in which you want to hide the global AS from the VPN.

Include the **no-prepend-global-as** option to have the global AS configured at the **[edit routing-options]** hierarchy level removed from the AS path sent to external peers. When you use this option, only the local AS is included in the AS path for the routes sent to a customer edge (CE) device.

- **Number of loops option**—The local AS feature also supports specifying the number of times that detection of the AS number in the AS_PATH attribute causes the route to be discarded or hidden. For example, if you configure **loops 1**, the route is hidden if the AS number is detected in the path one or more times. This is the default behavior. If you configure **loops 2**, the route is hidden if the AS number is detected in the path two or more times.

For the **loops number** statement, you can configure 1 through 10.



NOTE: If you configure the local AS values for any BGP group, the detection of routing loops is performed using both the AS and the local AS values for all BGP groups.

If the local AS for the EBGP or IBGP peer is the same as the current AS, do not use the `local-as` statement to specify the local AS number.

When you configure the local AS within a VRF, this impacts the AS path loop-detection mechanism. All of the `local-as` statements configured on the device are part of a single AS domain. The AS path loop-detection mechanism is based on looking for a matching AS present in the domain.

**Related
Documentation**

- [Example: Configuring a Local AS for EBGP Sessions on page 5](#)
- [Example: Configuring a Private Local AS for EBGP Sessions on page 15](#)

Example: Configuring a Local AS for EBGP Sessions

This example shows how to configure a local autonomous system (AS) for a BGP peer so that both the global AS and the local AS are used in BGP inbound and outbound updates.

- [Requirements on page 5](#)
- [Overview on page 5](#)
- [Configuration on page 6](#)
- [Verification on page 12](#)

Requirements

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

Overview

Use the **local-as** statement when ISPs merge and want to preserve a customer's configuration, particularly the AS with which the customer is configured to establish a peer relationship. The **local-as** statement simulates the AS number already in place in customer routers, even if the ISP's router has moved to a different AS.

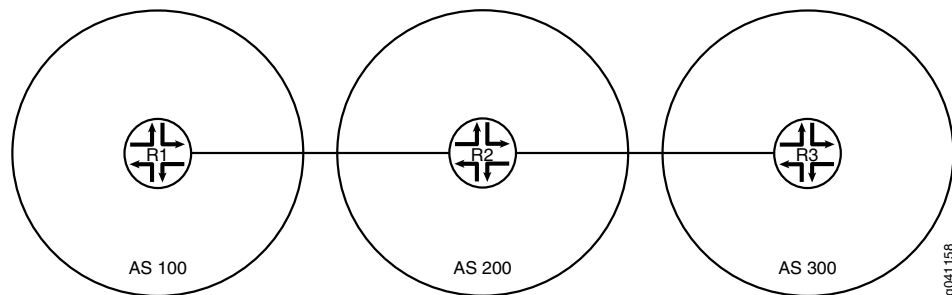
This example shows how to use the **local-as** statement to configure a local AS. The **local-as** statement is supported for BGP at the global, group, and neighbor hierarchy levels.

When you configure the **local-as** statement, you must specify an AS number. You can specify a number from 1 through 4,294,967,295 in plain-number format. In Junos OS Release 9.1 and later, the range for AS numbers is extended to provide BGP support for 4-byte AS numbers as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*. In Junos OS Release 9.3 and later, you can also configure a 4-byte AS number using the

AS-dot notation format of two integer values joined by a period: *<16-bit high-order value in decimal>.<16-bit low-order value in decimal>*. For example, the 4-byte AS number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format. You can specify a value from 0.0 through 65535.65535 in AS-dot notation format. Junos OS continues to support 2-byte AS numbers. The 2-byte AS number range is 1 through 65,535 (this is a subset of the 4-byte range).

Figure 2 on page 6 shows the sample topology.

Figure 2: Topology for Configuring the Local AS



In this example, Device R2 formerly belonged to AS 250 and now is in AS 200. Device R1 and Device R3 are configured to peer with AS 250 instead of with the new AS number (AS 200). Device R2 has the new AS number configured with the **autonomous-system 200** statement. To enable the peering sessions to work, the **local-as 250** statement is added in the BGP configuration. Because **local-as 250** is configured, Device R2 includes both the global AS (200) and the local AS (250) in its BGP inbound and outbound updates.

Configuration

- [Configuring Device R1 on page 7](#)
- [Configuring Device R2 on page 9](#)
- [Configuring Device R3 on page 11](#)

CLI Quick Configuration To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

```
Device R1  set interfaces fe-1/2/0 unit 1 family inet address 10.0.0.1/30
           set interfaces lo0 unit 1 family inet address 192.168.0.1/32
           set protocols bgp group ext type external
           set protocols bgp group ext export send-direct
           set protocols bgp group ext export send-static
           set protocols bgp group ext peer-as 250
           set protocols bgp group ext neighbor 10.0.0.2
           set policy-options policy-statement send-direct term 1 from protocol direct
           set policy-options policy-statement send-direct term 1 then accept
           set policy-options policy-statement send-static term 1 from protocol static
           set policy-options policy-statement send-static term 1 then accept
           set routing-options static route 10.1.0.0/30 next-hop 10.0.0.2
           set routing-options autonomous-system 100
```

Device R2

```
set interfaces fe-1/2/0 unit 2 family inet address 10.0.0.2/30
set interfaces fe-1/2/1 unit 3 family inet address 10.1.0.1/30
set interfaces lo0 unit 2 family inet address 192.168.0.2/32
set protocols bgp group ext type external
set protocols bgp group ext export send-direct
set protocols bgp group ext export send-static
set protocols bgp group ext local-as 250
set protocols bgp group ext neighbor 10.0.0.1 peer-as 100
set protocols bgp group ext neighbor 10.1.0.2 peer-as 300
set policy-options policy-statement send-direct term 1 from protocol direct
set policy-options policy-statement send-direct term 1 then accept
set policy-options policy-statement send-static term 1 from protocol static
set policy-options policy-statement send-static term 1 then accept
set routing-options autonomous-system 200
```

Device R3

```
set interfaces fe-1/2/0 unit 4 family inet address 10.1.0.2/30
set interfaces lo0 unit 3 family inet address 192.168.0.3/32
set protocols bgp group ext type external
set protocols bgp group ext export send-direct
set protocols bgp group ext export send-static
set protocols bgp group ext peer-as 250
set protocols bgp group ext neighbor 10.1.0.1
set policy-options policy-statement send-direct term 1 from protocol direct
set policy-options policy-statement send-direct term 1 then accept
set policy-options policy-statement send-static term 1 from protocol static
set policy-options policy-statement send-static term 1 then accept
set routing-options static route 10.0.0.0/30 next-hop 10.1.0.1
set routing-options autonomous-system 300
```

Configuring Device R1

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode in the CLI User Guide*.

To configure Device R1:

1. Configure the interfaces.

```
[edit interfaces]
user@R1# set fe-1/2/0 unit 1 family inet address 10.0.0.1/30

user@R1# set lo0 unit 1 family inet address 192.168.0.1/32
```
2. Configure external BGP (EBGP).

```
[edit protocols bgp group ext]
user@R1# set type external
user@R1# set export send-direct
user@R1# set export send-static
user@R1# set peer-as 250
user@R1# set neighbor 10.0.0.2
```
3. Configure the routing policy.

```
[edit policy-options]
user@R1# set policy-statement send-direct term 1 from protocol direct
```

```
user@R1# set policy-statement send-direct term 1 then accept
user@R1# set policy-statement send-static term 1 from protocol static
user@R1# set policy-statement send-static term 1 then accept
```

4. Configure a static route to the remote network between Device R2 and Device R3.

```
[edit routing-options]
user@R1# set static route 10.1.0.0/30 next-hop 10.0.0.2
```

5. Configure the global AS number.

```
[edit routing-options]
user@R1# set autonomous-system 100
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show policy-options**, **show protocols**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
fe-1/2/0 {
  unit 1 {
    family inet {
      address 10.0.0.1/30;
    }
  }
}
lo0 {
  unit 1 {
    family inet {
      address 192.168.0.1/32;
    }
  }
}

user@R1# show policy-options
policy-statement send-direct {
  term 1 {
    from protocol direct;
    then accept;
  }
}
policy-statement send-static {
  term 1 {
    from protocol static;
    then accept;
  }
}

user@R1# show protocols
bgp {
  group ext {
    type external;
    export [ send-direct send-static ];
    peer-as 250;
    neighbor 10.0.0.2;
  }
}
```

```
}  
user@R1# show routing-options  
static {  
    route 10.1.0.0/30 next-hop 10.0.0.2;  
}  
autonomous-system 100;
```

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

Configuring Device R2

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the CLI User Guide.

To configure Device R2:

1. Configure the interfaces.
[edit interfaces]
user@R2# set fe-1/2/0 unit 2 family inet address 10.0.0.2/30

user@R2# set fe-1/2/1 unit 3 family inet address 10.1.0.1/30

user@R2# set lo0 unit 2 family inet address 192.168.0.2/32
2. Configure EBGP.
[edit protocols bgp group ext]
user@R2# set type external
user@R2# set export send-direct
user@R2# set export send-static
user@R2# set neighbor 10.0.0.1 peer-as 100
user@R2# set neighbor 10.1.0.2 peer-as 300
3. Configure the local autonomous system (AS) number.
[edit protocols bgp group ext]
user@R2# set local-as 250
4. Configure the global AS number.
[edit routing-options]
user@R2# set autonomous-system 200
5. Configure the routing policy.
[edit policy-options]
user@R2# set policy-statement send-direct term 1 from protocol direct
user@R2# set policy-statement send-direct term 1 then accept
user@R2# set policy-statement send-static term 1 from protocol static
user@R2# set policy-statement send-static term 1 then accept

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show policy-options**, **show protocols**, and **show routing-options** commands. If the output

does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R2# show interfaces
fe-1/2/0 {
  unit 2 {
    family inet {
      address 10.0.0.2/30;
    }
  }
}
fe-1/2/1 {
  unit 3 {
    family inet {
      address 10.1.0.1/30;
    }
  }
}
lo0 {
  unit 2 {
    family inet {
      address 192.168.0.2/32;
    }
  }
}

user@R2# show policy-options
policy-statement send-direct {
  term 1 {
    from protocol direct;
    then accept;
  }
}
policy-statement send-static {
  term 1 {
    from protocol static;
    then accept;
  }
}

user@R2# show protocols
bgp {
  group ext {
    type external;
    export [ send-direct send-static ];
    local-as 250;
    neighbor 10.0.0.1 {
      peer-as 100;
    }
    neighbor 10.1.0.2 {
      peer-as 300;
    }
  }
}

user@R2# show routing-options
autonomous-system 200;
```

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

Configuring Device R3

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the CLI User Guide.

To configure Device R3:

1. Configure the interfaces.

```
[edit interfaces]
user@R3# set fe-1/2/0 unit 4 family inet address 10.1.0.2/30

user@R3# set lo0 unit 3 family inet address 192.168.0.3/32
```
2. Configure EBGP.

```
[edit protocols bgp group ext]
user@R3# set type external
user@R3# set export send-direct
user@R3# set export send-static
user@R3# set peer-as 250
user@R3# set neighbor 10.1.0.1
```
3. Configure the global autonomous system (AS) number.

```
[edit routing-options]
user@R3# set autonomous-system 300
```
4. Configure a static route to the remote network between Device R1 and Device R2.

```
[edit routing-options]
user@R3# set static route 10.0.0.0/30 next-hop 10.1.0.1
```
5. Configure the routing policy.

```
[edit policy-options]
user@R3# set policy-statement send-direct term 1 from protocol direct
user@R3# set policy-statement send-direct term 1 then accept
user@R3# set policy-statement send-static term 1 from protocol static
user@R3# set policy-statement send-static term 1 then accept
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show policy-options**, **show protocols**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R3# show interfaces
fe-1/2/0 {
  unit 4 {
    family inet {
      address 10.1.0.2/30;
    }
  }
}
```

```
lo0 {
  unit 3 {
    family inet {
      address 192.168.0.3/32;
    }
  }
}

user@R3# show policy-options
policy-statement send-direct {
  term 1 {
    from protocol direct;
    then accept;
  }
}
policy-statement send-static {
  term 1 {
    from protocol static;
    then accept;
  }
}

user@R3# show protocols
bgp {
  group ext {
    type external;
    export [ send-direct send-static ];
    peer-as 250;
    neighbor 10.1.0.1;
  }
}

user@R3# show routing-options
static {
  route 10.0.0.0/30 next-hop 10.1.0.1;
}
autonomous-system 300;
```

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

Verification

Confirm that the configuration is working properly.

- [Checking the Local and Global AS Settings on page 12](#)
- [Checking the BGP Peering Sessions on page 14](#)
- [Verifying the BGP AS Paths on page 14](#)

Checking the Local and Global AS Settings

Purpose Make sure that Device R2 has the local and global AS settings configured.

Action From operational mode, enter the **show bgp neighbors** command.

```
user@R2> show bgp neighbors
```



```

Peer: 10.0.0.1+179 AS 100      Local: 10.0.0.2+61036 AS 250
Type: External    State: Established    Flags: <Sync>
Last State: OpenConfirm    Last Event: RecvKeepAlive
Last Error: None
Export: [ send-direct send-static ]
Options: <Preference PeerAS LocalAS Refresh>
Holdtime: 90 Preference: 170 Local AS: 250 Local System AS: 200
Number of flaps: 0
Peer ID: 192.168.0.1      Local ID: 192.168.0.2      Active Holdtime: 90
Keepalive Interval: 30      Peer index: 0
BFD: disabled, down
Local Interface: fe-1/2/0.2
NLRI for restart configured on peer: inet-unicast
NLRI advertised by peer: inet-unicast
NLRI for this session: inet-unicast
Peer supports Refresh capability (2)
Stale routes from peer are kept for: 300
Peer does not support Restarter functionality
NLRI that restart is negotiated for: inet-unicast
NLRI of received end-of-rib markers: inet-unicast
NLRI of all end-of-rib markers sent: inet-unicast
Peer supports 4 byte AS extension (peer-as 100)
Peer does not support Addpath
Table inet.0 Bit: 10000
  RIB State: BGP restart is complete
  Send state: in sync
  Active prefixes:          1
  Received prefixes:        3
  Accepted prefixes:        2
  Suppressed due to damping: 0
  Advertised prefixes:      4
Last traffic (seconds): Received 6    Sent 14    Checked 47
Input messages:  Total 258    Updates 3      Refreshes 0      Octets 4969
Output messages: Total 258    Updates 2      Refreshes 0      Octets 5037
Output Queue[0]: 0

```

```

Peer: 10.1.0.2+179 AS 300      Local: 10.1.0.1+52296 AS 250
Type: External    State: Established    Flags: <Sync>
Last State: OpenConfirm    Last Event: RecvKeepAlive
Last Error: None
Export: [ send-direct send-static ]
Options: <Preference PeerAS LocalAS Refresh>
Holdtime: 90 Preference: 170 Local AS: 250 Local System AS: 200
Number of flaps: 0
Peer ID: 192.168.0.3      Local ID: 192.168.0.2      Active Holdtime: 90
Keepalive Interval: 30      Peer index: 1
BFD: disabled, down
Local Interface: fe-1/2/1.3
NLRI for restart configured on peer: inet-unicast
NLRI advertised by peer: inet-unicast
NLRI for this session: inet-unicast
Peer supports Refresh capability (2)
Stale routes from peer are kept for: 300
Peer does not support Restarter functionality
NLRI that restart is negotiated for: inet-unicast
NLRI of received end-of-rib markers: inet-unicast
NLRI of all end-of-rib markers sent: inet-unicast
Peer supports 4 byte AS extension (peer-as 300)
Peer does not support Addpath
Table inet.0 Bit: 10000
  RIB State: BGP restart is complete

```

```

Send state: in sync
Active prefixes:      1
Received prefixes:    3
Accepted prefixes:    2
Suppressed due to damping: 0
Advertised prefixes:  4
Last traffic (seconds): Received 19   Sent 26   Checked 9
Input messages: Total 256   Updates 3     Refreshes 0   Octets 4931
Output messages: Total 256   Updates 2     Refreshes 0   Octets 4999
Output Queue[0]: 0

```

Meaning The Local AS: 250 and Local System AS: 200 output shows that Device R2 has the expected settings. Additionally, the output shows that the options list includes LocalAS.

Checking the BGP Peering Sessions

Purpose Ensure that the sessions are established and that the local AS number 250 is displayed.

Action From operational mode, enter the **show bgp summary** command.

```

user@R1> show bgp summary
Groups: 1 Peers: 1 Down peers: 0
Table      Tot Paths  Act Paths Suppressed  History  Damp State   Pending
inet.0      4          2          0          0        0      0
Peer        AS      InPkt    OutPkt    OutQ   Flaps  Last Up/Dwn
State|#Active/Received/Accepted/Damped...
10.0.0.2      250      232      233       0        4    1:42:37
2/4/4/0      0/0/0/0

user@R3> show bgp summary
Groups: 1 Peers: 1 Down peers: 0
Table      Tot Paths  Act Paths Suppressed  History  Damp State   Pending
inet.0      4          2          0          0        0      0
Peer        AS      InPkt    OutPkt    OutQ   Flaps  Last Up/Dwn
State|#Active/Received/Accepted/Damped...
10.1.0.1      250      235      236       0        4    1:44:25
2/4/4/0      0/0/0/0

```

Meaning Device R1 and Device R3 appear to be peering with a device in AS 250, even though Device R2 is actually in AS 200.

Verifying the BGP AS Paths

Purpose Make sure that the routes are in the routing tables and that the AS paths show the local AS number 250.

Action From configuration mode, enter the **set route protocol bgp** command.

```

user@R1> show route protocol bgp
inet.0: 6 destinations, 8 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/30      [BGP/170] 01:46:44, localpref 100
                  AS path: 250 I
                  > to 10.0.0.2 via fe-1/2/0.1
10.1.0.0/30      [BGP/170] 01:46:44, localpref 100
                  AS path: 250 I

```

```

192.168.0.2/32      > to 10.0.0.2 via fe-1/2/0.1
                  *[BGP/170] 01:46:44, localpref 100
                  AS path: 250 I
192.168.0.3/32      > to 10.0.0.2 via fe-1/2/0.1
                  *[BGP/170] 01:46:40, localpref 100
                  AS path: 250 300 I
                  > to 10.0.0.2 via fe-1/2/0.1

user@R3> show route protocol bgp

inet.0: 6 destinations, 8 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/30          [BGP/170] 01:47:10, localpref 100
                    AS path: 250 I
                    > to 10.1.0.1 via fe-1/2/0.4
10.1.0.0/30          [BGP/170] 01:47:10, localpref 100
                    AS path: 250 I
                    > to 10.1.0.1 via fe-1/2/0.4
192.168.0.1/32       *[BGP/170] 01:47:10, localpref 100
                    AS path: 250 100 I
                    > to 10.1.0.1 via fe-1/2/0.4
192.168.0.2/32       *[BGP/170] 01:47:10, localpref 100
                    AS path: 250 I
                    > to 10.1.0.1 via fe-1/2/0.4

```

Meaning The output shows that Device R1 and Device R3 appear to have routes with AS paths that include AS 250, even though Device R2 is actually in AS 200.

Related Documentation

- 4-Byte Autonomous System Numbers Overview in the [Using 4-Byte Autonomous System Numbers in BGP Networks Technology Overview](#)
- Understanding External BGP Peering Sessions
- BGP Configuration Overview

Example: Configuring a Private Local AS for EBGP Sessions

This example shows how to configure a private local autonomous system (AS) number. The local AS is considered to be private because it is advertised to peers that use the local AS number for peering, but is hidden in the announcements to peers that can use the global AS number for peering.

- [Requirements on page 15](#)
- [Overview on page 16](#)
- [Configuration on page 17](#)
- [Verification on page 20](#)

Requirements

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

Overview

Use the **local-as** statement when ISPs merge and want to preserve a customer's configuration, particularly the AS with which the customer is configured to establish a peer relationship. The **local-as** statement simulates the AS number already in place in customer routers, even if the ISP's router has moved to a different AS.

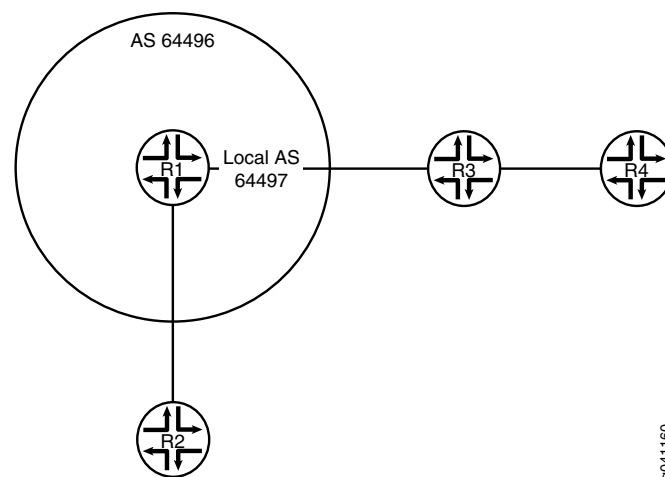
When you use the **private** option, the local AS is used during the establishment of the BGP session with an external BGP (EBGP) neighbor, but is hidden in the AS path sent to other EBGP peers. Only the global AS is included in the AS path sent to external peers.

The **private** option is useful for establishing local peering with routing devices that remain configured with their former AS or with a specific customer that has not yet modified its peer arrangements. The local AS is used to establish the BGP session with the EBGP neighbor, but is hidden in the AS path sent to external peers in another AS.

Include the **private** option so that the local AS is not prepended before the global AS in the AS path sent to external peers. When you specify the **private** option, the local AS is prepended only in the AS path sent to the EBGP neighbor.

Figure 3 on page 16 shows the sample topology.

Figure 3: Topology for Configuring a Private Local AS



Device R1 is in AS 64496. Device R2 is in AS 64510. Device R3 is in AS 64511. Device R4 is in AS 64512. Device R1 formerly belonged to AS 64497, which has merged with another network and now belongs to AS 64496. Because Device R3 still peers with Device R1, using its former AS, 64497, Device R1 needs to be configured with a local AS of 64497 in order to maintain peering with Device R3. Configuring a local AS of 64497 permits Device R1 to add AS 64497 when advertising routes to Device R3. Device R3 sees an AS path of 64497 64496 for the prefix 10.1.1.2/32, which is Device R2's loopback interface. Device R4, which is behind Device R3, sees an AS path of 64511 64497 64496 64510 to Device R2's loopback interface. To prevent Device R1 from adding the local AS number in its announcements to other peers, this example includes the **local-as 64497 private** statement. The **private** option configures Device R1 to not include the local AS 64497

when announcing routes to Device R2. Device R2 sees an AS path of 64496 64511 to Device R3 and an AS path of 64496 64511 64512 to Device R4. The **private** option in Device R1's configuration causes the AS number 64497 to be missing from the AS paths that Device R1 readvertises to Device R2.

Device R2 is hiding the private local AS from all the routers, except Device R3. The **private** option applies to the routes that Device R1 receives (learns) from Device R3 and that Device R1, in turn, readvertises to other routers. When these routes learned from Device R3 are readadvertised by Device R1 to Device R2, the private local AS is missing from the AS path advertised to Device R2.

Configuration

CLI Quick Configuration To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

Device R1

```
set interfaces fe-1/2/0 unit 3 family inet address 192.168.1.1/24
set interfaces fe-1/2/1 unit 5 family inet address 192.168.10.1/24
set interfaces lo0 unit 2 family inet address 10.1.1.1/32
set protocols bgp group external-AS64511 type external
set protocols bgp group external-AS64511 peer-as 64511
set protocols bgp group external-AS64511 local-as 64497
set protocols bgp group external-AS64511 local-as private
set protocols bgp group external-AS64511 neighbor 192.168.1.2
set protocols bgp group external-AS64510 type external
set protocols bgp group external-AS64510 peer-as 64510
set protocols bgp group external-AS64510 neighbor 192.168.10.2
set policy-options policy-statement send-direct term 1 from protocol direct
set policy-options policy-statement send-direct term 1 then accept
set routing-options autonomous-system 64496
```

Device R2

```
set interfaces fe-1/2/0 unit 6 family inet address 192.168.10.2/24
set interfaces lo0 unit 3 family inet address 10.1.1.2/32
set protocols bgp group external type external
set protocols bgp group external export send-direct
set protocols bgp group external peer-as 64496
set protocols bgp group external neighbor 192.168.10.1
set policy-options policy-statement send-direct term 1 from protocol direct
set policy-options policy-statement send-direct term 1 then accept
set routing-options autonomous-system 64510
```

Device R3

```
set interfaces fe-1/2/0 unit 4 family inet address 192.168.1.2/24
set interfaces fe-1/2/1 unit 7 family inet address 192.168.5.1/24
set interfaces lo0 unit 4 family inet address 10.1.1.3/32
set protocols bgp group external type external
set protocols bgp group external export send-direct
set protocols bgp group external neighbor 192.168.1.1 peer-as 64497
set protocols bgp group external neighbor 192.168.5.2 peer-as 64512
set policy-options policy-statement send-direct term 1 from protocol direct
set policy-options policy-statement send-direct term 1 then accept
set routing-options autonomous-system 64511
```

Device R4

```
set interfaces fe-1/2/0 unit 8 family inet address 192.168.5.2/24
set interfaces lo0 unit 5 family inet address 10.1.1.4/32
set protocols bgp group external type external
set protocols bgp group external export send-direct
set protocols bgp group external peer-as 64511
set protocols bgp group external neighbor 192.168.5.1
set policy-options policy-statement send-direct term 1 from protocol direct
set policy-options policy-statement send-direct term 1 then accept
set routing-options autonomous-system 64512
```

Configuring Device R1

Step-by-Step Procedure The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode in the CLI User Guide*.

To configure Device R1:

1. Configure the interfaces.

```
[edit interfaces fe-1/2/0 unit 3]
user@R1# set family inet address 192.168.1.1/24

[edit interfaces fe-1/2/1 unit 5]
user@R1# set family inet address 192.168.10.1/24

[edit interfaces lo0 unit 2]
user@R1# set family inet address 10.1.1.1/32
```
2. Configure the EBGP peering session with Device R2.

```
[edit protocols bgp group external-AS64510]
user@R1# set type external
user@R1# set peer-as 64510
user@R1# set neighbor 192.168.10.2
```
3. Configure the EBGP peering session with Device R3.

```
[edit protocols bgp group external-AS64511]
user@R1# set type external
user@R1# set peer-as 64511
user@R1# set local-as 64497
user@R1# set local-as private
user@R1# set neighbor 192.168.1.2
```
4. Configure the routing policy.

```
[edit policy-options policy-statement send-direct term 1]
user@R1# set from protocol direct
user@R1# set then accept
```
5. Configure the global autonomous system (AS) number.

```
[edit routing-options]
user@R1# set autonomous-system 64496
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show policy-options**, **show protocols**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
fe-1/2/0 {
  unit 3 {
    family inet {
      address 192.168.1.1/24;
    }
  }
}
fe-1/2/1 {
  unit 5 {
    family inet {
      address 192.168.10.1/24;
    }
  }
}
lo0 {
  unit 2 {
    family inet {
      address 10.1.1.1/32;
    }
  }
}

user@R1# show policy-options
policy-statement send-direct {
  term 1 {
    from protocol direct;
    then accept;
  }
}

user@R1# show protocols
bgp {
  group external-AS64511 {
    type external;
    peer-as 64511;
    local-as 64497 private;
    neighbor 192.168.1.2;
  }
  group external-AS64510 {
    type external;
    peer-as 64510;
    neighbor 192.168.10.2;
  }
}

user@R1# show routing-options
autonomous-system 64496;
```

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

Repeat the configuration as needed for the other devices in the topology.

Verification

Confirm that the configuration is working properly.

- [Checking Device R2's AS Paths on page 20](#)
- [Checking Device R3's AS Paths on page 20](#)

Checking Device R2's AS Paths

Purpose Make sure that Device R2 does not have AS 64497 in its AS paths to Device R3 and Device R4.

Action From operational mode, enter the **show route protocol bgp** command.

```
user@R2> show route protocol bgp
inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.3/32      *[BGP/170] 01:33:11, localpref 100
                  AS path: 64496 64511 I
                  > to 192.168.10.1 via fe-1/2/0.6
10.1.1.4/32      *[BGP/170] 01:33:11, localpref 100
                  AS path: 64496 64511 64512 I
                  > to 192.168.10.1 via fe-1/2/0.6
192.168.5.0/24  *[BGP/170] 01:49:15, localpref 100
                  AS path: 64496 64511 I
                  > to 192.168.10.1 via fe-1/2/0.6
```

Meaning Device R2's AS paths do not include AS 64497.

Checking Device R3's AS Paths

Purpose Make sure that Device R3 does not have AS 64497 in its AS path to Device R4.

Action From operational mode, enter the **show route protocol bgp** command.

```
user@R3> show route protocol bgp
inet.0: 7 destinations, 8 routes (7 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.2/32      *[BGP/170] 01:35:11, localpref 100
                  AS path: 64497 64496 64510 I
                  > to 192.168.1.1 via fe-1/2/0.4
10.1.1.4/32      *[BGP/170] 01:35:11, localpref 100
                  AS path: 64512 I
                  > to 192.168.5.2 via fe-1/2/1.7
192.168.5.0/24  [BGP/170] 01:51:15, localpref 100
                  AS path: 64512 I
                  > to 192.168.5.2 via fe-1/2/1.7
```

Meaning Device R3's route to Device R2 (prefix 10.1.1.2) includes both the local and the global AS configured on Device R1 (64497 and 64496, respectively).

-
- Related Documentation**
- [Understanding External BGP Peering Sessions](#)
 - [BGP Configuration Overview](#)

