Application Note

Basic Configuration Examples for BGP

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
408 745 2000 or 888 JUNIPER
www.juniper.net

Part Number: 350008-001 04/02

Contents

List of Figures

Purpose

This technical note describes the basic steps in configuring the Border Gateway Protocol (BGP) in the JUNOS software.

Terms

autonomous-system is the local AS of the router being configured.

group allows you to group or classify neighbors so that you can apply different import and export policies to them.

local-address tells BGP what source address to use when sending BGP packets to the configured neighbor. If it is not specified, the router's source address selection rules are used.

neighbor is the IP address of the remote peer.

peer-as is the AS number of the neighbor (peer).

router-id is the BGP identifier in BGP open messages. If it is not explicitly configured, the IP address of the first interface encountered for the router is used.

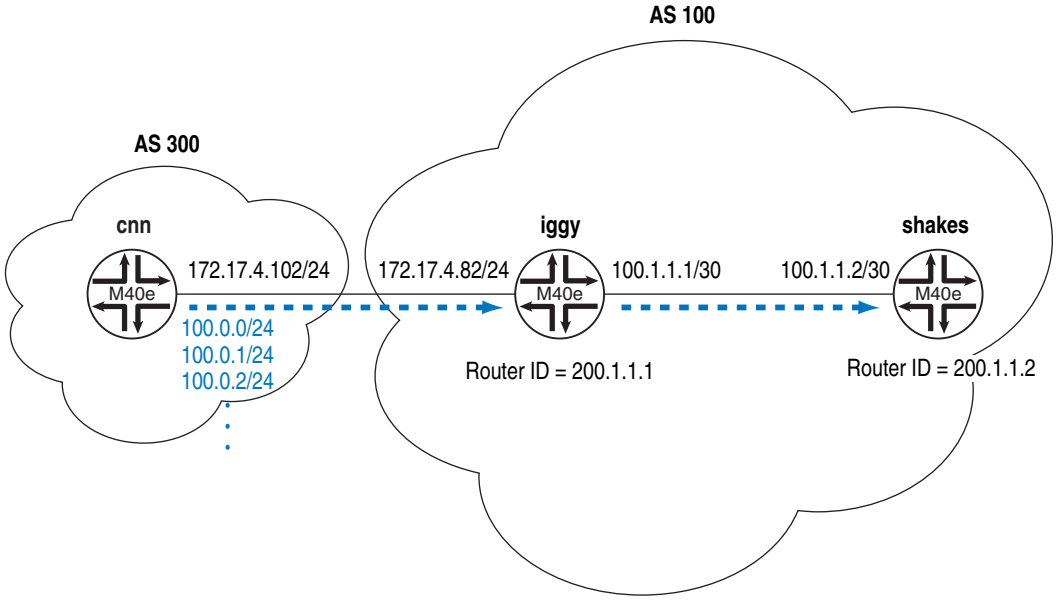
type indicates whether the neighbor is internal (IBGP peering) or external (EBGP peering) to the AS. If it is not specified, **type** defaults to external.

Setup

Before you configure BGP, you need to determine whether routing information will be exchanged between autonomous systems or within a single AS. If the exchange is between different autonomous systems, the peering session is an external BGP (EBGP) connection. If within a single AS, the session is an internal BGP (IBGP) connection.

In Figure 1, the routers cnn and iggy are EBGP peers, and the routers iggy and shakes are IBGP peers.

Figure 1: EGP and IBGP Peers



Let us examine how iggy and shakes are configured using the minimum required configuration statements.

iggy configuration

```
routing-options {
    static {
        route 0.0.0.0/0 {
            nexthop 172.17.4.254;
            retain;
            no-readvertise;
        }
    }
    router-id 200.1.1.1;
    autonomous-system 100;
}
protocols {
    bgp {
        group ibgp {
            type internal;
            peer-as 100;
            local-address 200.1.1.1;
            neighbor 200.1.1.2;
        }
        group ebgp {
            type external;
            peer-as 300;
            neighbor 172.17.4.102;
        }
    }
    ospf {
        area 0.0.0.0 {
            interface at-2/1/0.0;
            interface lo0.0 {
                passive;
            }
```

```
        }
    }
}
```

shakes configuration

```
routing-options {
    static {
        route 0.0.0.0/0 {
            nexthop 172.17.4.254;
            retain;
            no-readvertise;
        }
    }
    router-id 200.1.1.2;
    autonomous-system 100;
}
protocols {
    bgp {
        group ibgp {
            type internal;
            peer-as 100;
            local-address 200.1.1.2;
            neighbor 200.1.1.1;
        }
    }
    ospf {
        area 0.0.0.0 {
            interface at-0/1/1.0;
            interface lo0.0 {
                passive;
            }
        }
    }
}
```

In Figure 3, the neighbor address points to the loopback address of the remote peer. Peering with loopback addresses makes sense for IBGP because loopback addresses are always up. As long as there is a route to the loopback address, the IBGP peering session stays up. If the neighbor address is set to be the directly attached interface address, if the interface goes up and down, the IBGP peering session will flap up and down.

Because our neighbor is peering with the loopback address, the neighbor is expecting BGP messages to come from that address. Therefore, you need to include the **local-address** statement to configure that address. If you do not configure it, BGP updates are sourced using the router's source address selection rules (normally the directly connected interface address) and the peering session will not be established.

If the neighbor is set to the directly connected interface address, you can remove the **local-address** statement from the configuration so that the router uses the interface address as the source of BGP packets that it sends.

Default BGP Import and Export Behavior

In Figure 1, router cnn advertises routes 100.0.0/24, 100.0.1/24, 100.0.2/24, and so on to iggy. The default JUNOS software behavior is to import and export ALL BGP routes for both IBGP and EBGP peering. This behavior is illustrated by the following command output, which shows the following routes that are received and advertised by the router iggy.

iggy configuration

```
root@iggy> show route receive-protocol bgp 172.17.4.102

inet.0: 65547 destinations, 65547 routes (65546 active, 0 holddown, 1 hidden)
Prefix              Nexthop                 MED    Lclpref AS path
100.0.0.0/24        172.17.4.102                           300 I
100.0.1.0/24        172.17.4.102                           300 I
100.0.2.0/24        172.17.4.102                           300 I
100.0.3.0/24        172.17.4.102                           300 I
100.0.4.0/24        172.17.4.102                           300 I
100.0.5.0/24        172.17.4.102                           300 I
100.0.6.0/24        172.17.4.102                           300 I
100.0.7.0/24        172.17.4.102                           300 I
100.0.8.0/24        172.17.4.102                           300 I
100.0.9.0/24        172.17.4.102                           300 I
100.0.10.0/24       172.17.4.102                           300 I
```

```
root@iggy> show route advertising-protocol bgp 200.1.1.2

inet.0: 65547 destinations, 65547 routes (65546 active, 0 holddown, 1 hidden)
Prefix              Nexthop                 MED    Lclpref AS path
100.0.0.0/24        172.17.4.102                       100 300 I
100.0.1.0/24        172.17.4.102                       100 300 I
100.0.2.0/24        172.17.4.102                       100 300 I
100.0.3.0/24        172.17.4.102                       100 300 I
100.0.4.0/24        172.17.4.102                       100 300 I
100.0.5.0/24        172.17.4.102                       100 300 I
100.0.6.0/24        172.17.4.102                       100 300 I
100.0.7.0/24        172.17.4.102                       100 300 I
100.0.8.0/24        172.17.4.102                       100 300 I
100.0.9.0/24        172.17.4.102                       100 300 I
100.0.10.0/24       172.17.4.102                       100 300 I
```

If you want to filter certain routes, you need to apply routing policy (policies).

BGP Passive Operation

To have the router not send active BGP open messages to the neighbor, include the **passive** statement. The router instead waits for the peer to send an open message first before sending one.

You can specify the **passive** statement:

- Globally ˆ All BGP peers are passive
- For a specific group ˆ All BGP peers in that group are passive
- For a particular peer ˆ Only that BGP peer is passive

Example:

```
protocols {
   bgp {
      passive;                                  # global configuration
      group test {
            type internal;
            peer-as 100;
            local-address 200.1.1.1;
            passive;                            # group configuration
            neighbor 200.1.1.2 {
                passive;                        # neighbor configuration
              }
}
```

BGP Holdtime and Keepalive Timers

The BGP **holdtime** is a negotiated value and the `keepalive` is a calculated value equal to **one-third of the negotiated holdtime**. For the routers in Figure 1, we can configure the holdtime to 60 seconds on iggy and to 30 seconds on shakes.

iggy configuration

```
bgp {
    group ibgp {
        type internal;
        peer-as 100;
           hold-time 60;
        local-address 200.1.1.1;
        neighbor 200.1.1.2;
    }
}
```

shakes configuration

```
bgp {
    group ibgp {
        type internal;
        peer-as 100;
           hold-time 30;
        local-address 200.1.1.2;
        neighbor 200.1.1.1;
    }
}
```

This configuration results in the holdtime being negotiated to 30 seconds (the lower of the two values) and the keepalive being calculated to be 10 seconds (1/3 of 30 seconds). You can display these values in the output of the show bgp neighbor command.

```
root@shakes> show bgp neighbor

Peer: 100.1.1.1+179 AS 100      Local: 100.1.1.2+1052 AS 300
  Type: External  State: Established    Flags: <>
  Last State: OpenConfirm       Last Event: RecvKeepAlive
  Last Error: None
  Options: <Preference HoldTime Keepalive>
        Keepalive interval: 10  Holdtime: 30    Preference: 170
  Number of flaps: 0
  Peer ID: 200.1.1.1    Local ID: 200.1.1.2   Active Holdtime: 30
```



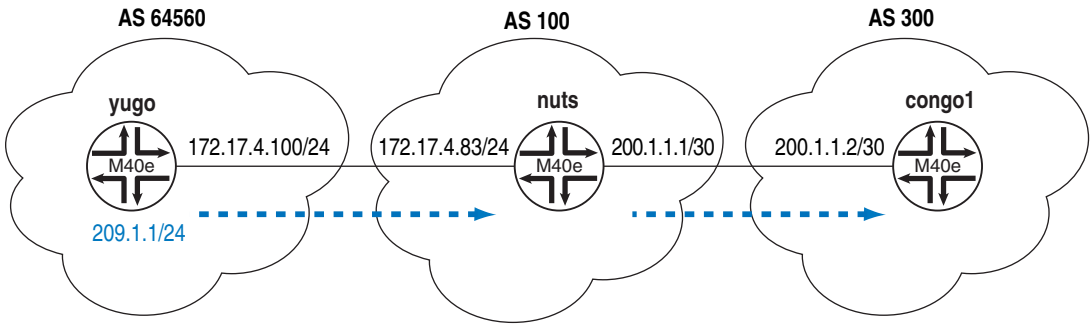
```
NLRI advertised by peer: unicast
NLRI for this session: unicast
Table inet.0 Bit: 1
  Active Prefixes: 0
  Received Prefixes: 0
  Suppressed due to damping: 0
Table inet.2 Bit: 10000
  Active Prefixes: 0
  Received Prefixes: 0
  Suppressed due to damping: 0
Last traffic (seconds):       Received 8      Sent 8  Checked 8
Input messages:       Total 1 Updates 0       Octets 19
Output messages:      Total 3 Updates 0       Octets 75
Output Queue[0]: 0
Output Queue[1]: 0
Trace options:  all
Trace file: /var/log/bgp.log  size 1048576    files 2
Route Queue Timer: unset      Route Queue: empty
```

BGP Remove Private Autonomous Systems

By default, BGP includes all AS numbers when advertising routes to its peers. The JUNOS BGP implementation allows the removal of private AS numbers from the AS path list. Private AS numbers are those in the range 64512 to 65535.

Figure 2 illustrates how to remove private AS numbers from the AS path.

Figure 2: Remove Private AS numbers from AS path.



The router called nuts in AS 100 receives the route 209.1.1/24 from an AS with a private AS number 64560. nuts then readvertises the route 209.1.1/24 to AS 300. The router congo1 in AS 300 receives the route with both AS numbers 64560 and 100 in the AS path information as shown in the following output.

```
root@congo1> show route protocol bgp detail
```

```
inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
209.1.1.0/24 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Nexthop: 200.1.1.1 via en0.0, selected
                State: <Active Ext>
                Local AS:   300 Peer AS:   100
                Age: 46
                Task: BGP_100.200.1.1.1+179
                Announcement bits (2): 1-KRT 2-BGP.0.0.0.0+179
```

```
                AS path: 100 64560 I
                Localpref: 100
                Router ID: 100.1.1.1
```

```
root@congo1> show route receive-protocol bgp 200.1.1.1 detail
```

```
inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
Prefix             Nexthop               MED    Lclpref AS path
209.1.1.0/24 (1 entry, 1 announced)
     Nexthop: 200.1.1.1
     AS path: 100 64560 I
```

To have the router in AS 100 remove private AS numbers from BGP updates to AS 300, specify the following configuration.

```
protocols {
    bgp {
        group ext1 {
            type external;
            peer-as 64560;
            neighbor 172.17.4.100;
        }
        group ext2 {
            type external;
            peer-as 300;
            remove-private;
            neighbor 200.1.1.2;
        }
    }
}
```

If you check congo1, you see that the private AS number 64560 has been removed from the AS path list.

```
root@congo1> show route protocol bgp detail
```

```
inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
209.1.1.0/24 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Nexthop: 200.1.1.1 via en0.0, selected
                State: <Active Ext>
                Local AS:   300 Peer AS:   100
                Age: 2:48
                Task: BGP_100.200.1.1.1+1051
                Announcement bits (2): 1-KRT 2-BGP.0.0.0.0+179
                AS path: 100 I
                Localpref: 100
                Router ID: 100.1.1.1
```

```
root@congo1> show route receive-protocol bgp 200.1.1.1 detail
```

```
inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
Prefix             Nexthop               MED    Lclpref AS path
209.1.1.0/24 (1 entry, 1 announced)
     Nexthop: 200.1.1.1
     AS path: 100 I
```

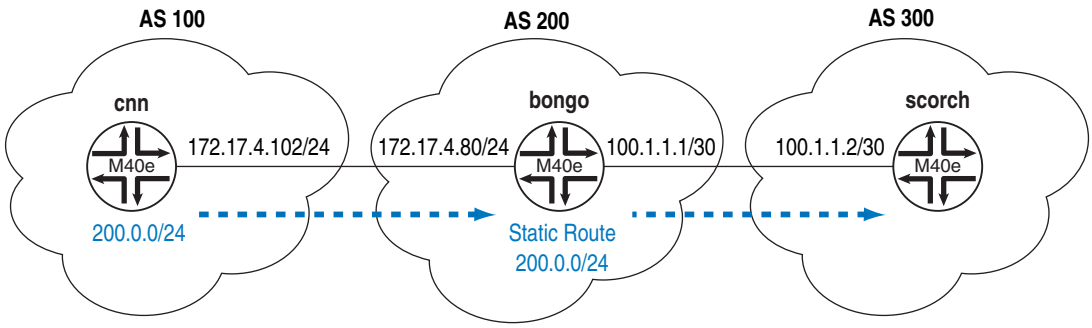
Just like the **passive** statement, the **remove-private** statement can be specified (1) globally, for a specific BGP group, or for a specific neighbor.

BGP Advertise Inactive Routes

By default, BGP advertises BGP routes that are installed or active, which are routes selected as the best based on the BGP path selection rules. The **advertise-inactive** statement allows nonactive BGP routes to be advertised to other peers. Figure 3 illustrates this example.

In this example, bongo has two EBGP peers, cnn and scorch. cnn advertises route 200.0.0/24 to bongo, but bongo has a manually configured static route for this prefix so it does not install the received BGP route.

Figure 3: BGP Advertise Inactive Routes



Following are the configuration and the output of the **show route** command on bongo:

```
root@bongo> show configuration
.
.
.
routing-options {
    static {
        route 0.0.0.0/0 {
            nexthop 172.17.4.254;
            retain;
            no-readvertise;
        }
        route 200.0.0.0/24 nexthop 172.17.4.101;
    }
    router-id 200.1.1.1;
    autonomous-system 200;
}
protocols {
    bgp {
        group ebgp2 {
            type external;
            peer-as 300;
            neighbor 100.1.1.2;
        }
        group ebgp {
            type external;
            peer-as 100;
            neighbor 172.17.4.102;
        }
    }
}
```

```
root@bongo> show route 200.0.0.0/24 detail

inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
200.0.0.0/24 (2 entries, 1 announced)
        *Static Preference:   5
                Nexthop: 172.17.4.101 via fxp0.0, selected
                State: <Active Int Ext>
                Age: 1:46
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I
         BGP    Preference: 170/-101
                Nexthop: 172.17.4.102 via fxp0.0, selected
                State: <Ext>
                Local AS:   200 Peer AS:   100
                Age: 2:42
                Task: BGP_100.172.17.4.102+1683
                AS path: 100 226 127 I
                Localpref: 100
                Router ID: 176.60.108.198
```

As you can see from the output, the static route is active and the BGP route is inactive. Because the BGP route is not active, bongo does not readvertise this BGP route to scorch. This is the default BGP behavior in the JUNOS software.

On bongo, the **show route receive-protocol** command shows the route being received from cnn, but the **show route advertising-protocol** command does not show any routes being readvertised to scorch.

```
root@bongo> show route receive-protocol bgp 172.17.4.102

inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
Prefix             Nexthop               MED    Lclpref AS path
200.0.0.0/24       172.17.4.102                          100 226 127 I

root@bongo> show route advertising-protocol bgp 100.1.1.2 detail
```

If you then add the **advertise-inactive** statement to our neighbor statement to scorch in bongo's configuration, bongo readvertises nonactive routes.

```
[edit protocols bgp group ebgp2]
root@bongo# show
type external;
peer-as 300;
neighbor 100.1.1.2 {
    advertise-inactive;
}

root@bongo> show route receive-protocol bgp 172.17.4.102 detail

inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
Prefix             Nexthop               MED    Lclpref AS path
200.0.0.0/24 (2 entries, 2 announced)
     Nexthop: 172.17.4.102
     AS path: 100 226 127 I

root@bongo> show route 200.0.0.0/24 detail

inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
200.0.0.0/24 (2 entries, 2 announced)
```



```
*Static Preference:   5
        Nexthop: 172.17.4.101 via fxp0.0, selected
        State: <Active Int Ext>
        Age: 5:07
        Task: RT
        Announcement bits (1): 0-KRT
        AS path: I
 BGP    Preference: 170/-101
        Nexthop: 172.17.4.102 via fxp0.0, selected
        State: <Ext>
        Local AS:   200 Peer AS:   100
        Age: 6:03
        Task: BGP_100.172.17.4.102+1683
        Announcement bits (1): 1-BGP.0.0.0.0+179
        AS path: 100 226 127 I
        Localpref: 100
        Router ID: 176.60.108.198
```

```
root@bongo> show route advertising-protocol bgp 100.1.1.2 detail

inet.0: 8 destinations, 8 routes (7 active, 0 holddown, 1 hidden)
Prefix             Nexthop               MED    Lclpref AS path
200.0.0.0/24 (2 entries, 2 announced)
 BGP 100.1.1.2 (External AS 300)
     Nexthop: Self
     AS path: 100 226 127 I
```

Additional Notes

When you are trying to debug BGP problems, it is helpful to enable BGP tracing operations to log important messages into a file. You can configure different traceoptions flags to filter the information that is logged.

In this example, the user configured BGP traceoptions to log BGP keepalive messages.

```
protocols {
    bgp {
        traceoptions {
            file bgp.log replace size 1m files 2;
            flag keepalive;
        }
```

For a list of the different traceoptions flags, refer to the JUNOS software configuration guide.

References

BGP Configuration Guidelines in JUNOS Internet Software Configuration Guide.