Application Note

JUNOS BGP Route Reflection
Configuration and Operation
Purpose

This technical note describes the configuration and operation of BGP route reflection in the JUNOS Internet software.

Requirements

One method to work around the need to have a full-mesh Internal BGP (IBGP) network is to use BGP route reflection. Before we discuss the route reflection configuration and operation in the JUNOS, it is important to understand the meaning of common terms used in this type of network design.

Terms

Cluster ID – The cluster ID is a configured four-octet value that identifies a route reflector and its clients.

Nonclient peer – Nonclient peers are routers that have a BGP session with the RR, but are not considered clients. Unlike RR clients, nonclient peers in the same autonomous system (AS) as the RR must be fully meshed.

Route reflector (RR) – The RR is the router responsible for the re-advertisement or the reflection of routes.

Router reflector client (RR client) – The RR client is the router that relies on the RR to advertise and to learn routes to and from the rest of the network. RR clients of the same RR might or might not be fully meshed.

Setup

In Figure 1, routers Iggy, Scorch and Roam are all in cluster 2.2.2.2. Iggy is the RR and Scorch and Roam are its RR clients. Router Congo3, a nonclient peer, has an IBGP session with Iggy.

In addition to Scorch being an RR client of Iggy, it is also serving as the RR in cluster 3.3.3.3 with Paul as its RR client.
The following is the BGP configuration on Iggy.

```plaintext
protocols {
    bgp {
        export static-to-bgp;
        group ibgp {
            type internal;
            local-address 2.2.2.2;
            peer-as 100;
            neighbor 5.5.5.5;
        }
        group rrcluster {
            type internal;
            local-address 2.2.2.2;
            cluster 2.2.2.2;
            peer-as 100;
            neighbor 1.1.1.1;
            neighbor 3.3.3.3;
        }
    }
}
```

The cluster statement indicates that Iggy is a route reflector for this BGP group, and the neighbor statements define the two RR clients, which are neighbors 1.1.1.1 and neighbor 3.3.3.3.
The following is the BGP configuration on Scorch.

```plaintext
protocols {
  bgp {
    group ibgp {
      type internal;
      local-address 3.3.3.3;
      peer-as 100;
      neighbor 2.2.2.2;
    }
    group rrclient {
      type internal;
      local-address 3.3.3.3;
      cluster 3.3.3.3;
      peer-as 100;
      neighbor 4.4.4.4;
    }
  }
}
```

If you look at the `group ibgp` portion of Scorch’s configuration, you will notice that it doesn’t even know that it is a client of Iggy. That is, RR client configuration is the same as a normal IBGP configuration.

The `group rrclient` statement configures Scorch as a RR of cluster 3.3.3.3 with router Paul (neighbor 4.4.4.4) as its client.

**Operation**

When a RR receives a route, it re-advertises the route using the following rules:

- If the RR receives a route from a nonclient peer, it reflects the route to all clients.
- If the RR receives a route from a client peer, it reflects the route to all nonclients and all client peers except the client who originated the route. (Note: If you configure the parameter `no-client-reflect`, the RR does not reflect routes to other clients. Include this statement only when all the clients are fully meshed.)

If the RR receives a route from an EBGP peer, it reflects the route to all client and nonclient peers.

In addition to these re-advertisement rules, the RR sets two BGP attributes.

1. Originator ID indicates the router ID of the router that originated the route into BGP.
2. Cluster list records the sequence of clusters that the route has traversed.

The RR uses the following rules when setting these attributes:

- A RR never adds the cluster ID, cluster list, and originator ID to a route that it received from an EBGP peer or to a route learned from another protocol other than IBGP (such as static route) while reflecting.
- If the reflection is client to client or client to nonclient, the cluster ID, cluster list, and originator ID are added to the route. An originator ID is created only if one is not set. The current cluster ID is added to the existing cluster list or a new cluster list is created if one is not present.
Once set, the originator ID and cluster list attributes are checked to prevent routing loops. The following routing loop checks are implemented in the JUNOS Internet software.

- If a route is received from a client in a cluster that has our own cluster ID (for the same cluster), the route is discarded and not installed in the routing table.
- If a route is received from a nonclient that has our own cluster ID, the route is accepted because it might have to be reflected to another cluster if the router is supporting multiple clusters.
- Do not reflect a path back to the originator of the route.
- Do not reflect a path to a client if the cluster ID of the cluster to which the client belongs is included in the cluster list.
- Discard any route received if the originator ID is the same as our router ID.

**Analysis**

Looking again at Figure 1. Look at three routes on the different routers.

1. **Route 200.1.1/24**
   
   (a) Congo3 advertises the route to Iggy using IBGP.

   **On Iggy:**

   
   
<table>
<thead>
<tr>
<th>Route 200.1.1.0/24 (1 entry, 1 announced)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preference: 170/-101</td>
</tr>
<tr>
<td>Source: 5.5.5.5</td>
</tr>
<tr>
<td>Nexthop: 104.1.1.1 via at-0/0/0.0, selected</td>
</tr>
<tr>
<td>State: &lt;Active Int Ext&gt;</td>
</tr>
<tr>
<td>Local AS: 100 Peer AS: 100</td>
</tr>
<tr>
<td>Age: 7:03 Metric2: 1</td>
</tr>
<tr>
<td>Task: BGP_100.5.5.5.5+179</td>
</tr>
<tr>
<td>Announcement bits (3): 0-KRT 4-BGP.0.0.0.0+179</td>
</tr>
<tr>
<td>5-BGP.Sync.Any</td>
</tr>
<tr>
<td>AS path: I</td>
</tr>
<tr>
<td>BGP next hop: 5.5.5.5</td>
</tr>
<tr>
<td>Localpref: 100</td>
</tr>
<tr>
<td>Router ID: 5.5.5.5</td>
</tr>
</tbody>
</table>

   **(b) Iggy adds the cluster ID 2.2.2.2 to the cluster list, sets the originator ID to 5.5.5.5, and then reflects it to Scorch and Roam.**

   **On Scorch:**

   
<table>
<thead>
<tr>
<th>Route 200.1.1.0/24 (1 entry, 1 announced)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preference: 170/-101</td>
</tr>
<tr>
<td>Source: 2.2.2.2</td>
</tr>
<tr>
<td>Nexthop: 101.1.1.1 via at-5/2/1.0, selected</td>
</tr>
<tr>
<td>State: &lt;Active Int Ext&gt;</td>
</tr>
<tr>
<td>Local AS: 100 Peer AS: 100</td>
</tr>
<tr>
<td>Age: 4:56 Metric2: 2</td>
</tr>
<tr>
<td>Task: BGP_100.2.2.2.2+179</td>
</tr>
<tr>
<td>Announcement bits (3): 0-KRT 4-BGP.0.0.0.0+179</td>
</tr>
<tr>
<td>5-BGP.Sync.Any</td>
</tr>
<tr>
<td>AS path: I &lt;Originator&gt;</td>
</tr>
<tr>
<td>Cluster list: 2.2.2.2</td>
</tr>
<tr>
<td>Originator ID: 5.5.5.5</td>
</tr>
<tr>
<td>BGP next hop: 5.5.5.5</td>
</tr>
<tr>
<td>Localpref: 100</td>
</tr>
<tr>
<td>Router ID: 2.2.2.2</td>
</tr>
</tbody>
</table>
On Roam:

200.1.1.0/24 (1 entry, 1 announced)  
*BGP Preference: 170/-101  
Source: 2.2.2.2  
Nexthop: 100.1.1.2 via at-1/3/0.0, selected  
State: <Active Int Ext>  
Local AS: 100 Peer AS: 100  
Age: 14 Metric2: 2  
Task: BGP_100.2.2.2+179  
Announcement bits (2): 0-KRT 5-BGP_Sync_Any  
AS path: I <Originator>  
Cluster list: 2.2.2.2  
Originator ID: 5.5.5.5  
BGP next hop: 5.5.5.5  
Localpref: 100  
Router ID: 2.2.2.2

(c) Scorch adds the cluster ID 3.3.3.3 to the cluster list and reflects the route to Paul.

On Paul:

200.1.1.0/24 (1 entry, 1 announced)  
*BGP Preference: 170/-101  
Source: 3.3.3.3  
Nexthop: 102.1.1.1 via at-2/2/0.0, selected  
State: <Active Int Ext>  
Local AS: 100 Peer AS: 100  
Age: 5:59 Metric2: 4  
Task: BGP_100.3.3.3+1487  
Announcement bits (2): 2-KRT 5-BGP_Sync_Any  
AS path: I <Originator>  
Cluster list: 3.3.3.3 2.2.2.2  
Originator ID: 5.5.5.5  
BGP next hop: 5.5.5.5  
Localpref: 100  
Router ID: 3.3.3.3

2. Route 201.1.1/24

(a) Iggy injects the static route into BGP.

On Iggy:

protocols {
  bgp {
    export static-to-bgp;
  }
}
policy-options {
  policy-statement static-to-bgp {
    from {
      protocol static;
      route-filter 201.1.1.0/24 exact;
    }
    then accept;
  }
}
(b) Iggy advertises the route to Congo3, Scorch and Roam. It does not add an originator ID and cluster information because this information is learned from the protocol static statement in the static-to-bgp policy.

On Congo3:

```
201.1.1.0/24 (1 entry, 1 announced)
*BGP    Preference: 170/-101
    Source: 2.2.2.2
    Nexthop: 104.1.1.2 via en0.0, selected
    State: <Active Int Ext>
    Local AS:   100 Peer AS:   100
    Age: 3:02       Metric2: 2
    Task: BGP_100.2.2.2+1038
    Announcement bits (2): 0-KRT 5-BGP_Sync_Any
    AS path: I
    BGP next hop: 2.2.2.2
    Localpref: 100
    Router ID: 2.2.2.2
```

On Scorch:

```
201.1.1.0/24 (1 entry, 1 announced)
*BGP    Preference: 170/-101
    Source: 2.2.2.2
    Nexthop: 101.1.1.1 via at-5/2/1.0, selected
    State: <Active Int Ext>
    Local AS:   100 Peer AS:   100
    Age: 4:56       Metric2: 2
    Task: BGP_100.2.2.2+179
    Announcement bits (3): 0-KRT 4-BGP.0.0.0.0+179
    5-BGP_Sync_Any
    AS path: I
    BGP next hop: 2.2.2.2
    Localpref: 100
    Router ID: 2.2.2.2
```

On Roam:

```
201.1.1.0/24 (1 entry, 1 announced)
*BGP    Preference: 170/-101
    Source: 2.2.2.2
    Nexthop: 100.1.1.2 via at-1/3/0.0, selected
    State: <Active Int Ext>
    Local AS:   100 Peer AS:   100
    Age: 14 Metric2: 2
    Task: BGP_100.2.2.2+179
    Announcement bits (2): 0-KRT 5-BGP_Sync_Any
    AS path: I
    BGP next hop: 2.2.2.2
    Localpref: 100
    Router ID: 2.2.2.2
```
(c) Scorch adds cluster ID 3.3.3.3 to the cluster list, sets originator ID to 2.2.2.2, and then reflects the route to Paul.

On Paul:

```
201.1.1.0/24 (1 entry, 1 announced)
  *BGP    Preference: 170/-101
  Source: 3.3.3.3
  Nexthop: 102.1.1.1 via at-2/2/0.0, selected
  State: <Active Int Ext>
  Local AS: 100  Peer AS: 100
  Age: 5:59    Metric2: 4
  Task: BGP_100.3.3.3.3+1487
  Announcement bits (2): 2-KRT 5-BGP_Sync_Any
  AS path: I <Originator>
  Cluster list: 3.3.3.3
  Originator ID: 2.2.2.2
  BGP next hop: 2.2.2.2
  Localpref: 100
  Router ID: 3.3.3.3
```

3. Route 202.1.1/24

(a) Paul advertises the route to Scorch.

On Scorch:

```
202.1.1.0/24 (1 entry, 1 announced)
  *BGP    Preference: 170/-101
  Source: 4.4.4.4
  Nexthop: 102.1.1.2 via at-5/2/0.0, selected
  State: <Active Int Ext>
  Local AS: 100  Peer AS: 100
  Age: 3:10    Metric2: 2
  Task: BGP_100.4.4.4.4+179
  Announcement bits (3): 0-KRT 4-BGP.0.0.0.0+179
  AS path: I
  BGP next hop: 4.4.4.4
  Localpref: 100
  Router ID: 4.4.4.4
```

(b) Scorch adds the cluster ID 3.3.3.3 to the cluster list, sets the originator ID to 4.4.4.4 and reflects the route to Iggy.

On Iggy:

```
202.1.1.0/24 (1 entry, 1 announced)
  *BGP    Preference: 170/-101
  Source: 3.3.3.3
  Nexthop: 101.1.1.2 via at-6/1/1.0, selected
  State: <Active Int Ext>
  Local AS: 100  Peer AS: 100
  Age: 2:25    Metric2: 3
```
**Task: BGP_100.3.3.3+1488**

**Announcement bits (3): 0-KRT 4-BGP.0.0.0.0+179**

5-BGP_Sync_Any

**AS path:** I <Originator>

**Cluster list:** 3.3.3.3

**Originator ID:** 4.4.4.4

**BGP next hop:** 4.4.4.4

**Localpref:** 100

**Router ID:** 3.3.3.3

(c) Iggy adds the cluster ID 2.2.2.2 to the cluster list and reflects the route to Congo3 and Roam.

On Congo3:

**202.1.1.0/24** (1 entry, 1 announced)

*BGP*  

**Preference:** 170/-101  
**Source:** 2.2.2.2  
**Nexthop:** 104.1.1.2 via en0.0, selected  
**State:** <Active Int Ext>  
**Local AS:** 100  
**Peer AS:** 100  
**Age:** 1:15  
**Metric2:** 4  
**Task:** BGP_100.2.2.2+1038  

**Announcement bits (2): 0-KRT 5-BGP_Sync_Any**

**AS path:** I <Originator>

**Cluster list:** 2.2.2.2 3.3.3.3

**Originator ID:** 4.4.4.4

**BGP next hop:** 4.4.4.4

**Localpref:** 100

**Router ID:** 2.2.2.2

On Roam:

**202.1.1.0/24** (1 entry, 1 announced)

*BGP*  

**Preference:** 170/-101  
**Source:** 2.2.2.2  
**Nexthop:** 100.1.1.2 via at-1/3/0.0, selected  
**State:** <Active Int Ext>  
**Local AS:** 100  
**Peer AS:** 100  
**Age:** 14  
**Metric2:** 4  
**Task:** BGP_100.2.2.2.2+179  

**Announcement bits (2): 0-KRT 5-BGP_Sync_Any**

**AS path:** I <Originator>

**Cluster list:** 2.2.2.2 3.3.3.3

**Originator ID:** 4.4.4.4

**BGP next hop:** 4.4.4.4

**Localpref:** 100

**Router ID:** 2.2.2.2
In Figure 2, router Congo3 now belongs to a different AS (AS 200).

Figure 2: BGP Route Reflection Advertisement within an EBGP Configuration

The route advertisements are the same as the example above in Figure 1, except that now Iggy does not set the originator ID and the cluster list attributes when it reflects the route to Scorch and Roam because the route is now received from EBGP.

On Scorch:

200.1.1.0/24 (1 entry, 1 announced)
  *BGP  Preference: 170/-101
  Source: 2.2.2.2
  Nexthop: 101.1.1.1 via at-5/2/1.0, selected
  State: <Active Int Ext>
  Local AS: 100 Peer AS: 100
  Age: 2:17 Metric2: 2
  Task: BGP_100.2.2.2+179
  Announcement bits (3): 0-KRT 4-BGP.0.0.0.0+179
  5-BGP_Sync_Any
  AS path: 200 I
  BGP next hop: 104.1.1.1
  Localpref: 100
  Router ID: 2.2.2.2

On Roam:

200.1.1.0/24 (1 entry, 1 announced)
  *BGP  Preference: 170/-101
  Source: 2.2.2.2
  Nexthop: 100.1.1.2 via at-1/3/0.0, selected
  State: <Active Int Ext>
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. To do this, include the `traceoptions` statement under BGP. You can configure different `traceoptions` flags to filter the information that is logged.

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

```plaintext
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 Internet Software Configuration Guide.

References

Configure Route Reflection in the BGP Configuration Guidelines of the JUNOS Internet Software Configuration Guide.

RFC 1966 (BGP Route Reflection: An alternative to full mesh IBGP)