

BGP Scaling (RR & Peer Group)

BGP Peer Group

- Problem: number of BGP updates in a iBGP mesh
 - BGP updates generated for each neighbor individually
 - CPU wasted on repeat calculations
 - iBGP neighbors receive the same update
 - Contain same info
- Solution: Peer Groups
 - Group neighbors with the same outbound update policy
 - Updates are generated once per group

BGP Peer Group

- Still need to establish TCP sessions individually
- Useful when many neighbors have the same outbound policies
 - Runs through the outbound filters only once for the group (applied to all members)
- Members can have a different inbound policy!
- Simplifies configuration
 - Define the peer group
 - Add neighbors to the peer group
 - Still need to configure peering individually
 - Apply filters (outbound) to the group

Peer Group – Best Practices

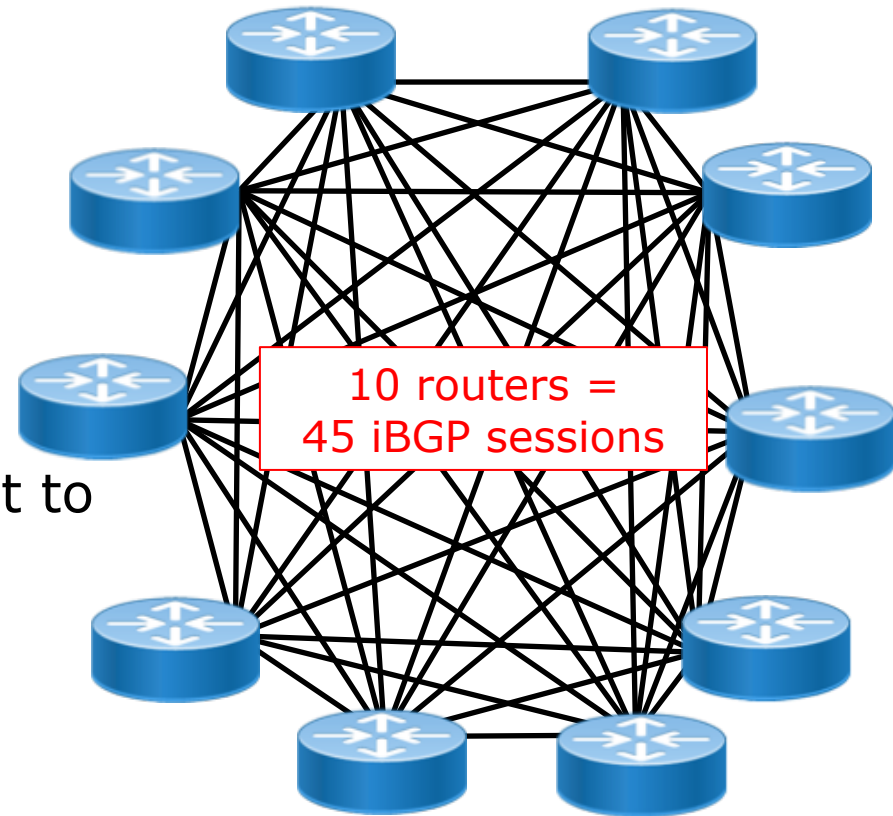
- Always configure peer-groups for iBGP
 - Even if there are only a few iBGP peers
 - Easier to scale network in the future
- Consider using peer-groups for eBGP
 - Especially useful for multiple BGP customers using same AS (RFC 2270)
 - Also at IXPs where ISP policy is generally the same for each IX peer

BGP Loop Prevention

- eBGP
 - AS-PATH attribute
 - If the local ASN is seen in a route received from a eBGP peer, a routing loop has occurred
 - Drop the route!
- iBGP
 - BGP router is not allowed to advertise iBGP learned routes to other iBGP peers within the AS
 - How do all iBGP routers learn about each other's networks?
 - iBGP full-mesh!

Scaling iBGP mesh

- Number of iBGP sessions
 $n(n-1)/2$
 - 10 routers = 45 sessions
 - 100 routers = 4950 sessions
- Number of BGP updates
 - Every update needs to be sent to all iBGP peer



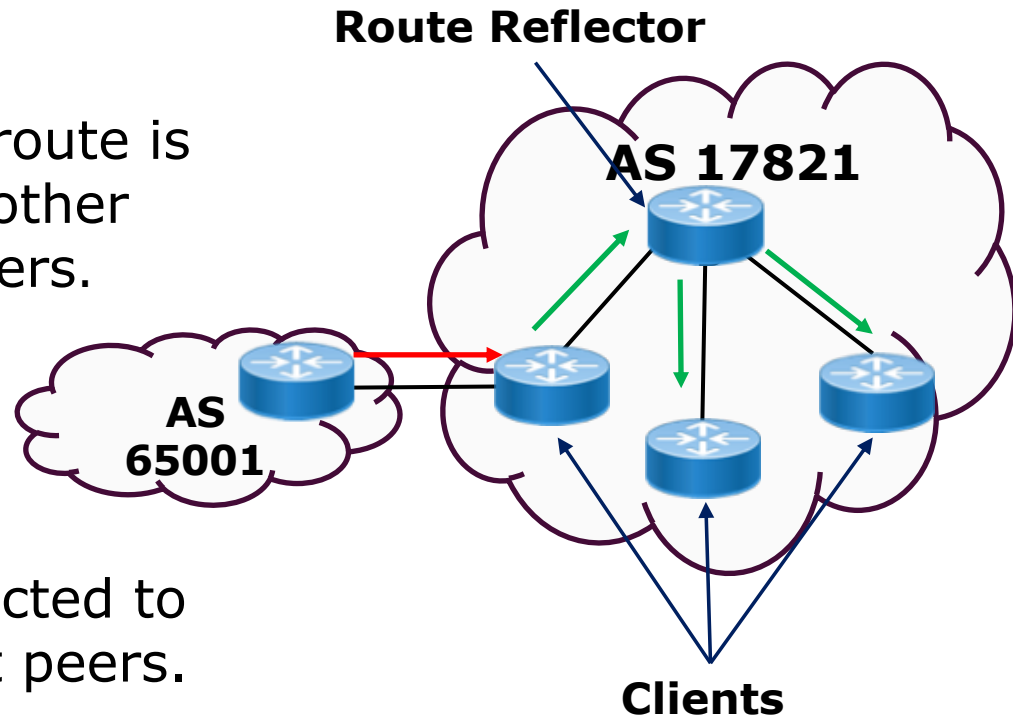
Solution

- **Route Reflection** (RFC4456)
 - RR client peers only with the RR
 - RR and its clients form a CLUSTER
 - Non-clients
 - An AS can have multiple clusters

RRs do not affect the actual traffic path;
only affects the path BGP messages take!

RR Operation

- When a RR receives an Update:
 - If from a client peer, the route is reflected (advertised) to other clients, and non-client peers.
 - If from a non-client, only reflected to client peers
 - If from a eBGP peer, reflected to both client and non-client peers.



Routing loops can happen with RRs!

Avoiding loops in RR

- **Originator_ID** attribute
 - The BGP router id of the originator, created by the RR
 - If you see your router id in the Originator_ID attribute, loop has occurred.
- **Cluster_List** attribute
 - In a cluster with a single RR, the Cluster ID is the router ID of the RR
 - If more than one RR in a cluster, a 4-byte Cluster ID configured
 - Cluster_List reflects the sequence of clusters a route has passed through
 - When a RR reflects routes (from clients) to non-clients, it appends the local Cluster ID to the Cluster_List
 - When a RR receives an update, if the local Cluster ID is seen in the list, a loop has occurred! (drops the update)

RR Design

- Divide the AS into multiple clusters
 - At least one RR and few clients in a cluster
 - Could have more than one RR in a cluster for redundancy
 - **NOT recommended!**
- Peering between clients in a cluster not necessary (but could be)
 - The RR reflects routes between them
- RRs in different clusters must be fully meshed with each other and with any iBGP router that's not a part of any cluster

RR example

- RR configuration

```
router bgp 17821
  <output-omitted>
  neighbor 2406:6400:2 remote-as 17821
  neighbor 2406:6400:2 route-reflector-client
  neighbor 2406:6400:3 remote-as 17821
  neighbor 2406:6400:3 route-reflector-client
  neighbor 2406:6400:4 remote-as 17821
  neighbor 2406:6400:4 route-reflector-client
<output-omitted>
```

- RR Client config

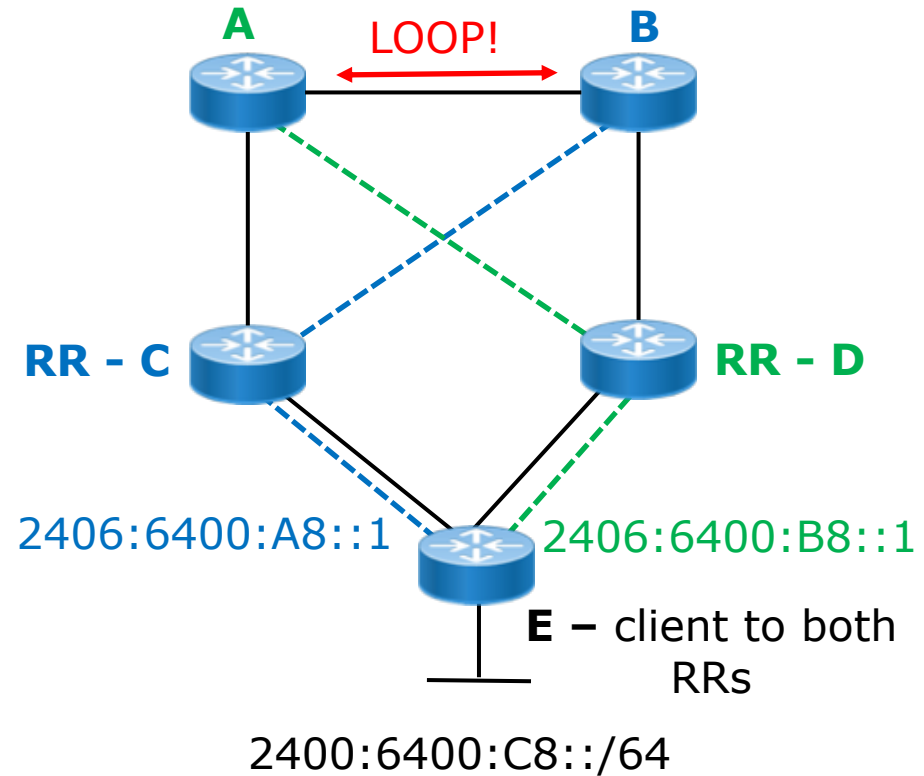
```
router bgp 17821
  <output-omitted>
  neighbor 2406:6400:1 remote-as 17821
  <output-omitted>
```



Only peers with the RR

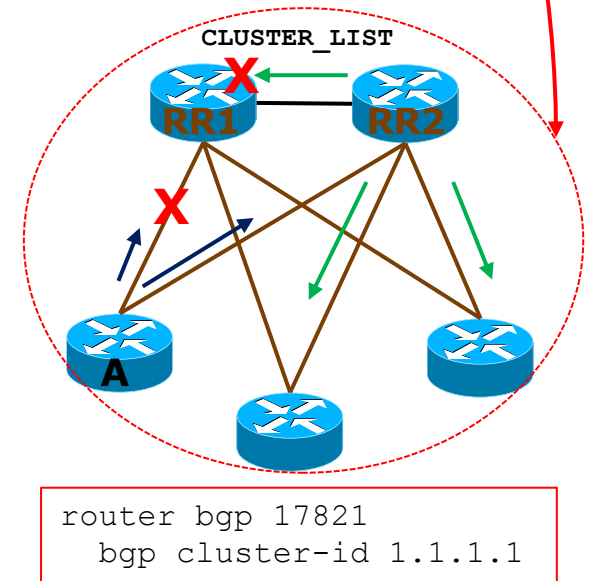
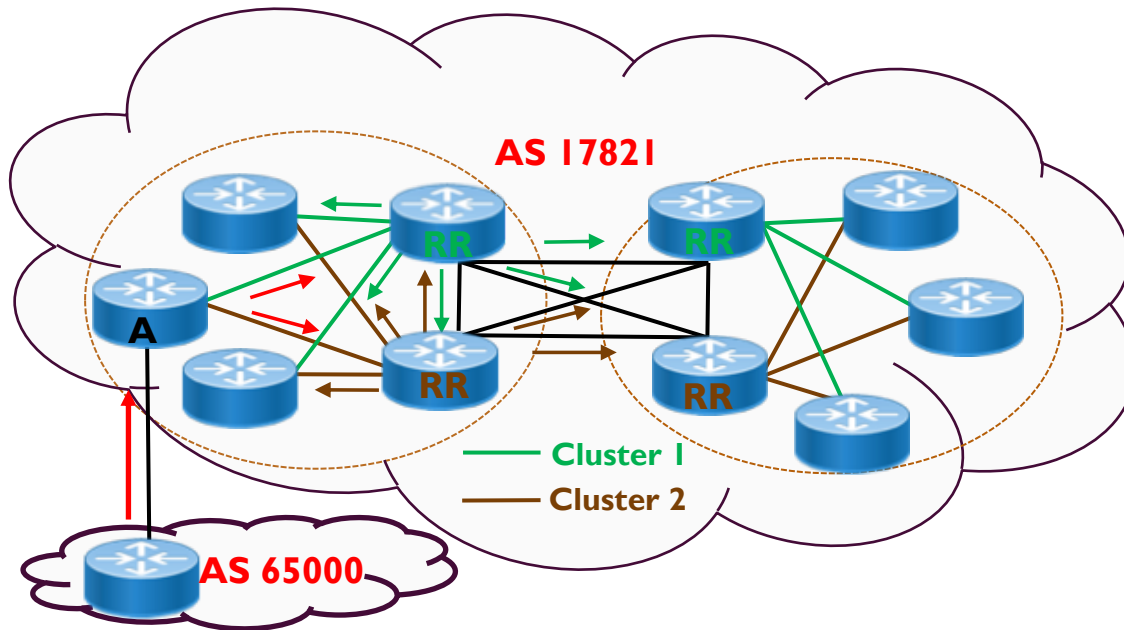
RR Selection

- Best practice is to follow the physical topology
 - Ensures traffic forwarding paths wont be affected
 - Prevents routing loops



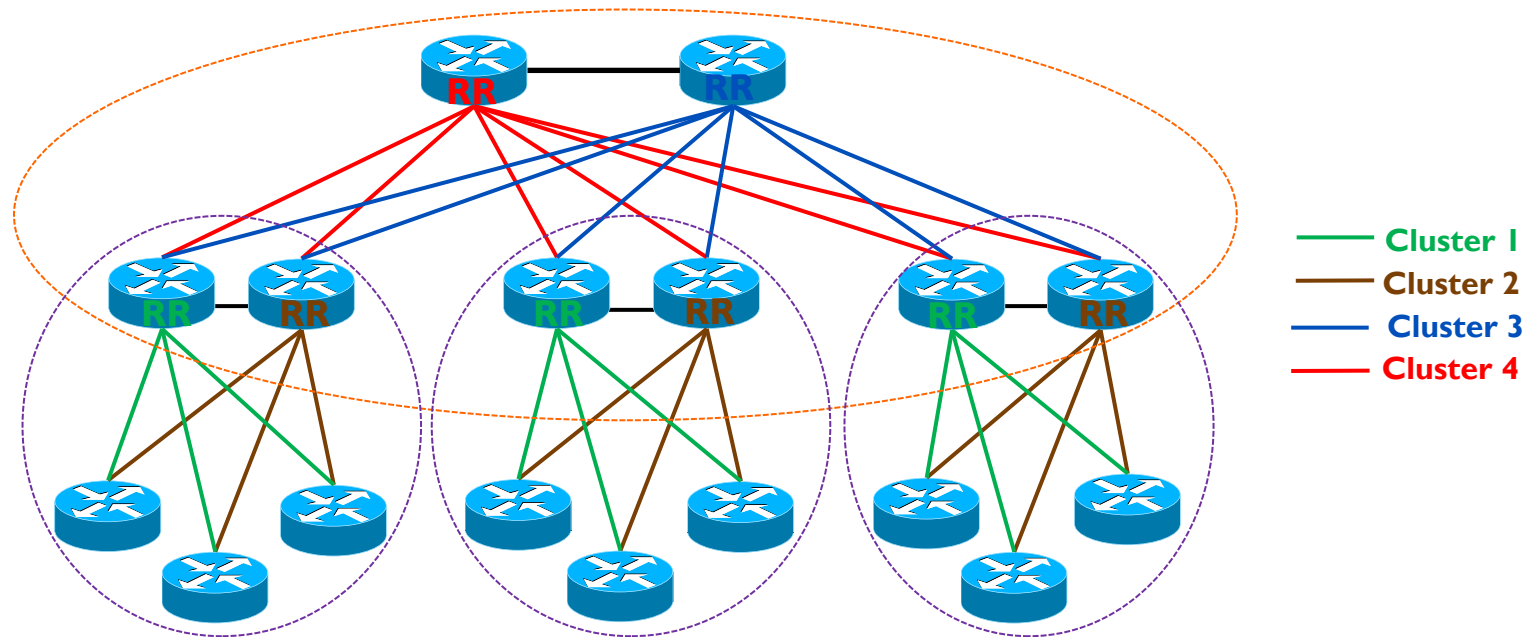
RR Redundancy

- Most ISP networks would overlay two clusters
 - Each client peers with RRs in different clusters (same POP) for redundancy (**NEVER** two RRs in the same cluster!)
 - All RRs fully-meshed!
 - Can have full-mesh between clients in the same cluster



RR Redundancy – Best Practice

- A hierarchical RR design
 - RRs of some clusters are clients of other RRs in different clusters
 - Less iBGP sessions
 - Easier to manage and scale





Questions

