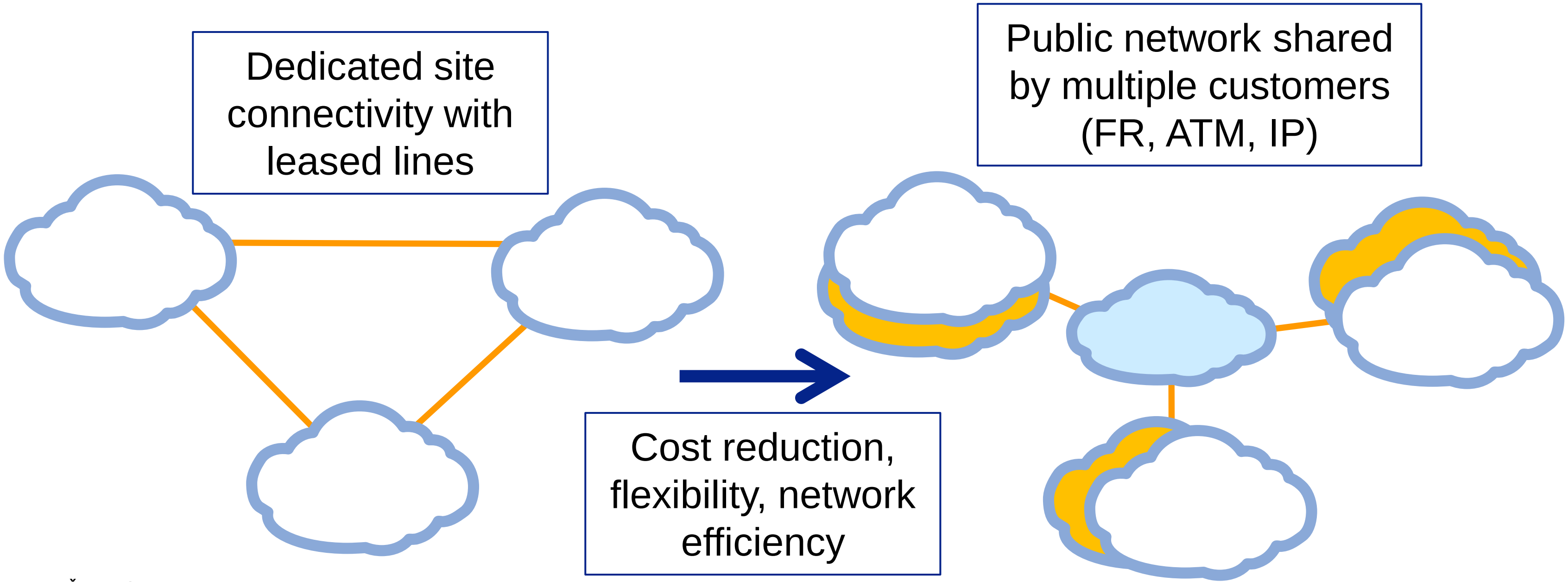
L3 VPN

Virtual Private Networks (VPN)

- *Virtual*
  - Emulated connectivity over a public network
- *Private*
  - Access limited to VPN members
  - Total address and route separation
- *Network*
  - A collection of customer sites



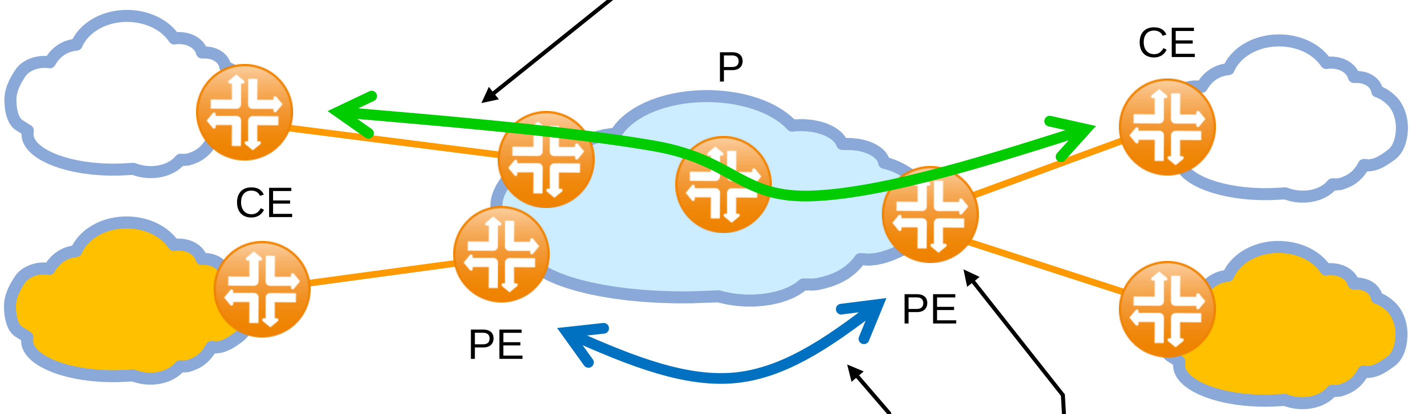
Dedicated site connectivity with leased lines

Public network shared by multiple customers (FR, ATM, IP)

Cost reduction, flexibility, network efficiency

Type of IP VPNs

- *Classification based on where VPN functions are implemented*
  - Customer Edge (CE) – based VPN
  - Provider Edge (PE) – based VPN
- *Classification based on SP’s role in provisioning the VPN*
  - Provider Provisioned VPN (PPVPN)
  - Customer Provisioned VPN



CE-CE tunnels (IPIP, IPSec, GRE) in CE-based VPNs over IP/MPLS net

PE-PE tunnels (IP, IPSec, GRE, MPLS) in PE-based VPNs

PE maps CE traffic to interfaces

Type of VPN Based on Protocol Layer

Layer 2 PE VPNs

- Provider network (PE routers) switches customer Layer-2 frames based on Layer-2 header
- P routers deliver layer 2 circuits to the customer, one for each remote site
- PE routers map customer’s layer 3 routing to the circuit mesh
- Customer routes are transparent to provider

Layer 3 PE VPNs

- SP network routes incoming customer packets based on the destination IP address
- SP network participates in customer’s layer 3 routing
- SP network manages VPN-specific routing tables, distributes routes to remote sites
- CPE routers advertise their routes to the provider

How does it work?

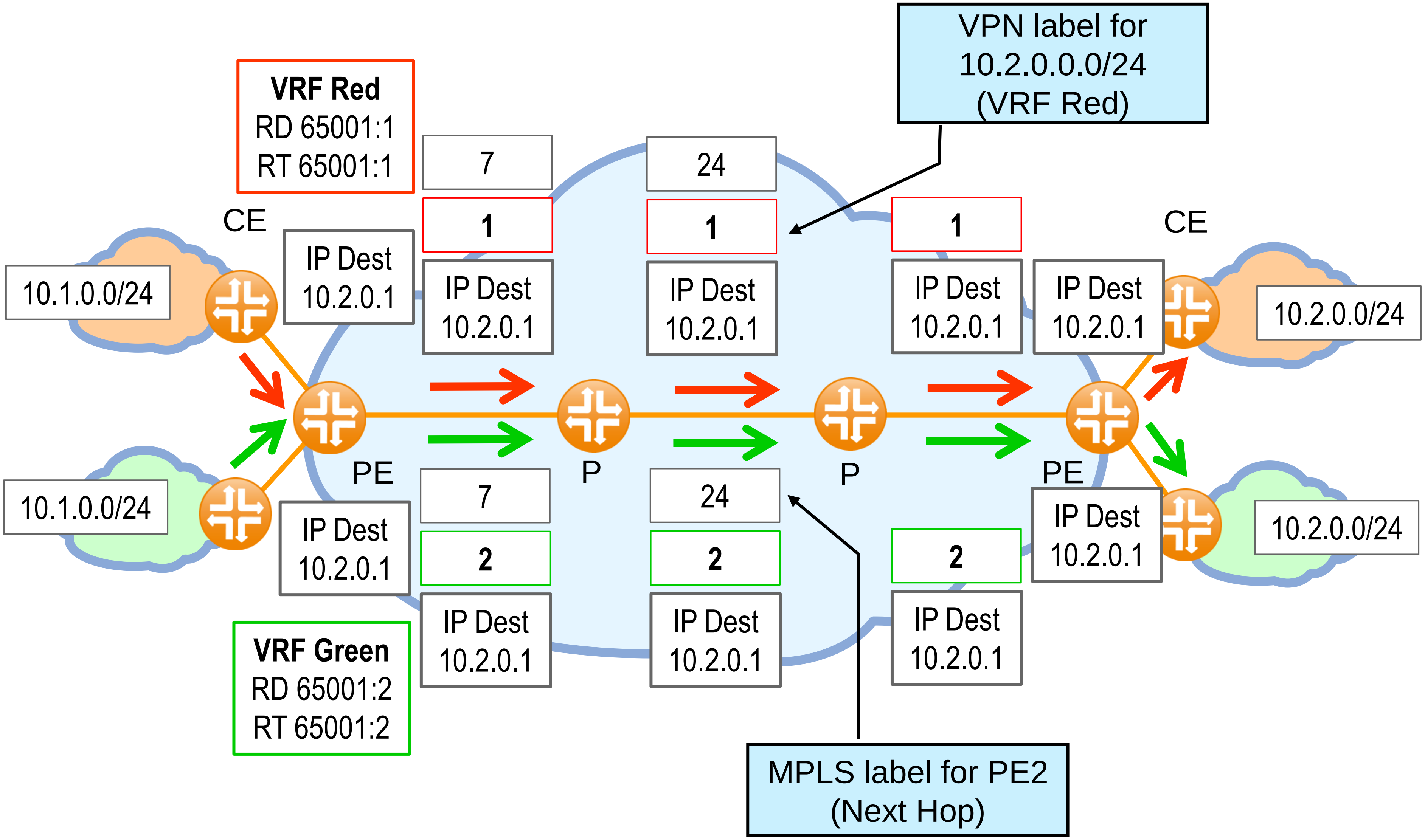
- The VPN PE marks packets from CE routers with two labels - the **inner label** is for the destination VPN route and the second **outside label** selects the label switched path to the remote VPN PE (BGP next hop) that originated the destination VPN route

- **VRF**s (Virtual Routing and Forwarding instances) are defined on the VPN PE
  - Each VRF instance represents the end point of a VPN (a separate routing table, a set of interfaces to which CEs are attached and CE/PE routing protocols (static/RIP/eBGP)

- To make VPN routes unique on a VPN-PE, the VRF needs to define a **Route Distinguisher**
  - RD pre-pended to each VPN route to make a VPN IPv4 route

- **Route Target** is a BGP community used for tagging prefixes
  - When exporting prefixes from the VRF, we add to the prefixes a RT, so the remote site can easily identify which prefixes to import

Data Forwarding in MPLS VPN Network



Additional attributes to BGP to Carry MPLS-VPN Info

- PE routers maintain separate routing tables
  - Global routing table - contains all internal PE and P routes
  - VRF routing tables - associated with one or more directly VPN customer connected sites (CE routers)
- Routing protocols must have a mean to differentiate between routes with identical IP address prefixes but in different VPNs
  - BGP Multi Protocol Extensions allow BGP (MP-BGP) to carry routes from multiple address families
  - Additional attributes/parameters to get the VRF routing information off the PE and to other PEs
    - VPNv4 address family
    - RD: Route Distinguisher
    - RT: Route Target
    - Label

VPNv4 Address Family

- To control policy about who sees what routes
- In BGP for IP, 32-bit address + 32-bit mask makes a unique announcement
- In BGP for MPLS-VPN, 64-bit RD + 32-bit address + 32-bit mask makes a unique announcement
- Since the route encoding is different, need a different address family in BGP
- VPNv4 announcement carries a label with the route

Route Distinguisher

- To differentiate 10/8 in VPN-A from 10/8 in VPN-B
- 64-bit quantity (2 bytes type, 6 bytes value)
- Configured as ASN:YY or IPADDR:YY
  - Almost everybody uses ASN
- Purely to make a route unique
  - Unique route is now RD:IPAddr (96 bits) plus a mask on the IPAddr portion
  - So customers don’t see each others routes

Route Target

- To control policy about who sees what routes

- 64-bit quantity (2 bytes type, 6 bytes value)

- Carried as an extended community

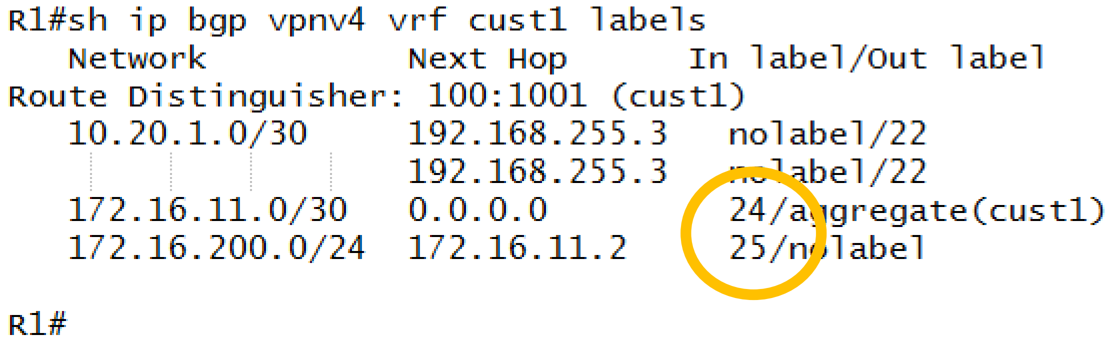
- Typically written as ASN:YY

- Each VRF ‘imports’ and ‘exports’ one or more RTs
  - Exported RTs are carried in VPNv4 BGP
  - Imported RTs are local to the box

- A PE that imports an RT installs that route in its routing table

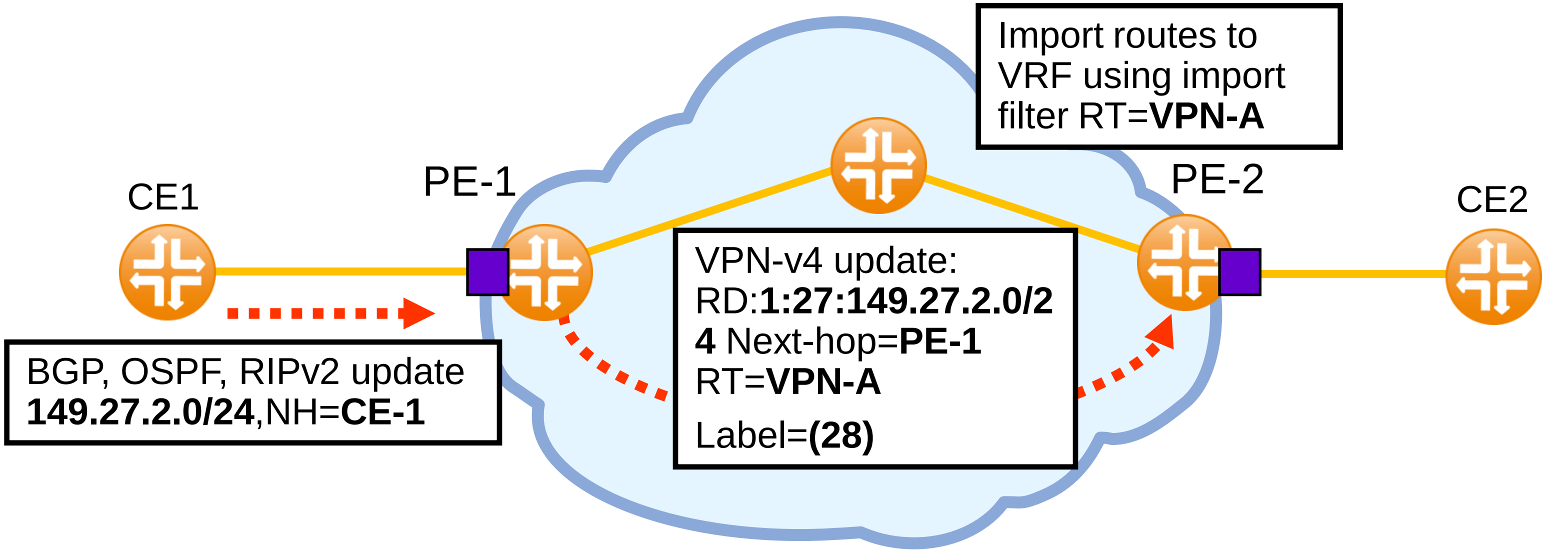
Inner Label

- Downstream PEs assign and distribute MPLS labels with the routes
  - A single label may be used for the entire VRF
  - A single label may be used for each attachment circuit
  - Different labels may be used for each route



```
R1#sh ip bgp vpnv4 vrf cust1 labels
   Network          Next Hop      In label/Out label
Route Distinguisher: 100:1001 (cust1)
   10.20.1.0/30     192.168.255.3   nolabel/22
                    192.168.255.3   nolabel/22
   172.16.11.0/30   0.0.0.0         24/aggregate(cust1)
   172.16.200.0/24  172.16.11.2     25/nolabel

R1#
```

VRF Population of MP-BGP


Import routes to VRF using import filter RT=**VPN-A**
CE1
PE-1
PE-2
CE2
VPN-v4 update: RD:**1:27:149.27.2.0/24** Next-hop=**PE-1** RT=**VPN-A**
Label=**(28)**
BGP, OSPF, RIPv2 update **149.27.2.0/24**,NH=**CE-1**

- PE router translates into VPN-V4 route
  - Assigns an RD and RT based on configuration
  - Re-writes Next-Hop attribute (PE loopback), assigns an inner label
  - Sends MP-BGP update to all PE neighbours
- Receiving PE routers translate to IPv4
  - Insert the route into the VRF identified by the RT
  - attribute (based on PE configuration)
- The label associated to the VPN-V4 address will be set on packets forwarded back towards the destination CE1

Multi Protocol BGP (MP-BGP)

- Defined in IETF RFC 4760, is an extension to BGP that allows different types of addresses (known as address families) to be distributed in parallel
  - IPv4 Unicast
  - IPv4 Multicast
  - IPv6 Unicast

- Few new attributes:
  - MP_REACH_NLRI - the set of reachable destinations
  - MP_UNREACH_NLRI - the set of unreachable destinations
  - Extended communities

- May be carried in same BGP session
- Same path selection and validation rules
  - AS-Path, LocalPref, MED, etc
- Separate BGP tables maintained

MP_REACH_NLRI Attribute

The key characteristics of this new attribute:

- Address Family Identifier (AFI) and Sub-AFI fields
- The Next-Hop Address information is contained in the field following the AFI
- Following the Next-Hop Address fields are zero or more SNPA (Sub Network Point Attachment) fields. These field contain the attributes associated with the NLRI field
    - Typically zero
- Finally, the NLRI field contains the Length, Prefix information, RD and Label of the route that is being advertised as reachable

Address Family Information

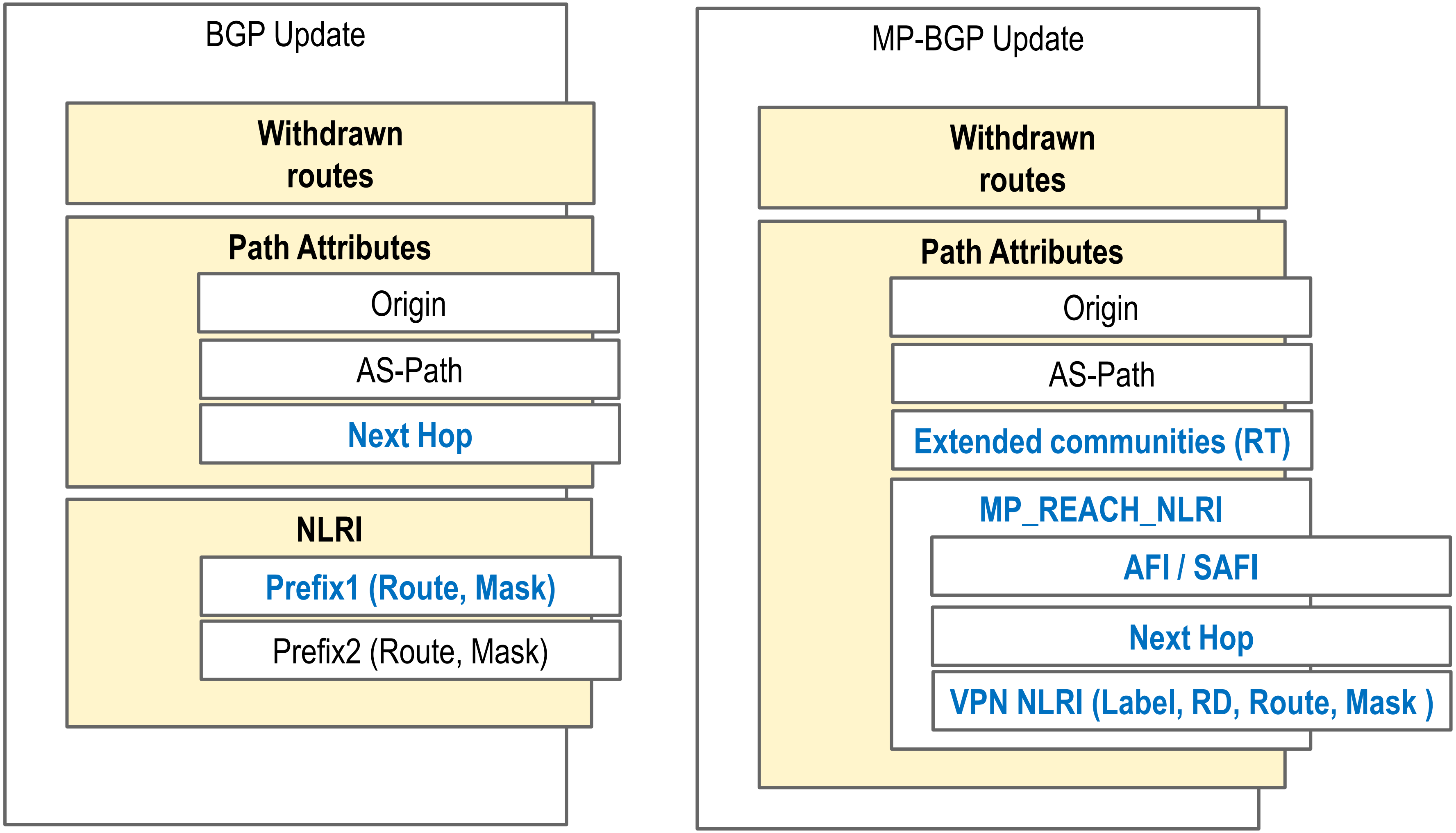
- Address Family Information (AFI)
  - Identifies Address Type (see RFC1700)
    - http://www.iana.org/assignments/address-family-numbers/address-family-numbers.xml
    - AFI = 1 (IPv4)
    - AFI = 2 (IPv6)
- Sub-Address Family Information (SAFI)
  - Sub category for AFI Field
    - http://www.iana.org/assignments/safi-namespace/safi-namespace.xml
  - Address Family Information (AFI) = 1 (IPv4)
    - Sub-AFI = 1 – NLRI used for unicast forwarding
    - Sub-AFI = 2 – NLRI used for multicast forwarding
    - Sub-AFI = 128 - MPLS-labeled VPN address

Extended communities

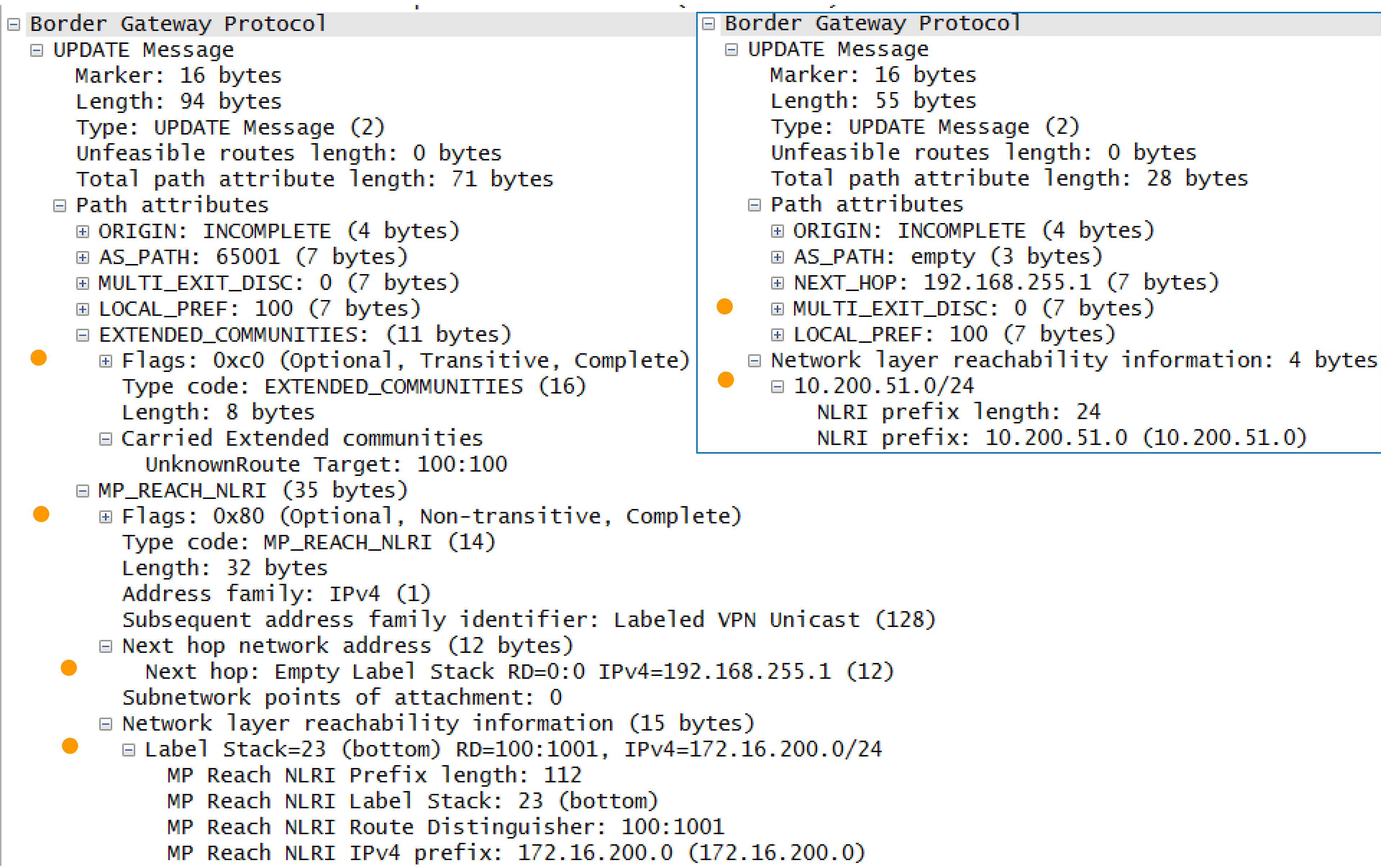
- RT - Route Target BGP Extended Community
  - Used for route filtering, already mentioned

- SOO - Site of Origin BGP Extended Community
  - The SOO prevents routing loops when a site is multihomed to the MPLS/VPN backbone
  - In addition, that site uses the AS-override feature achieved by identifying the site from where the route was learned, based on its SOO, so that it is not re-advertised back

Comparison BGP and MP-BGP Update



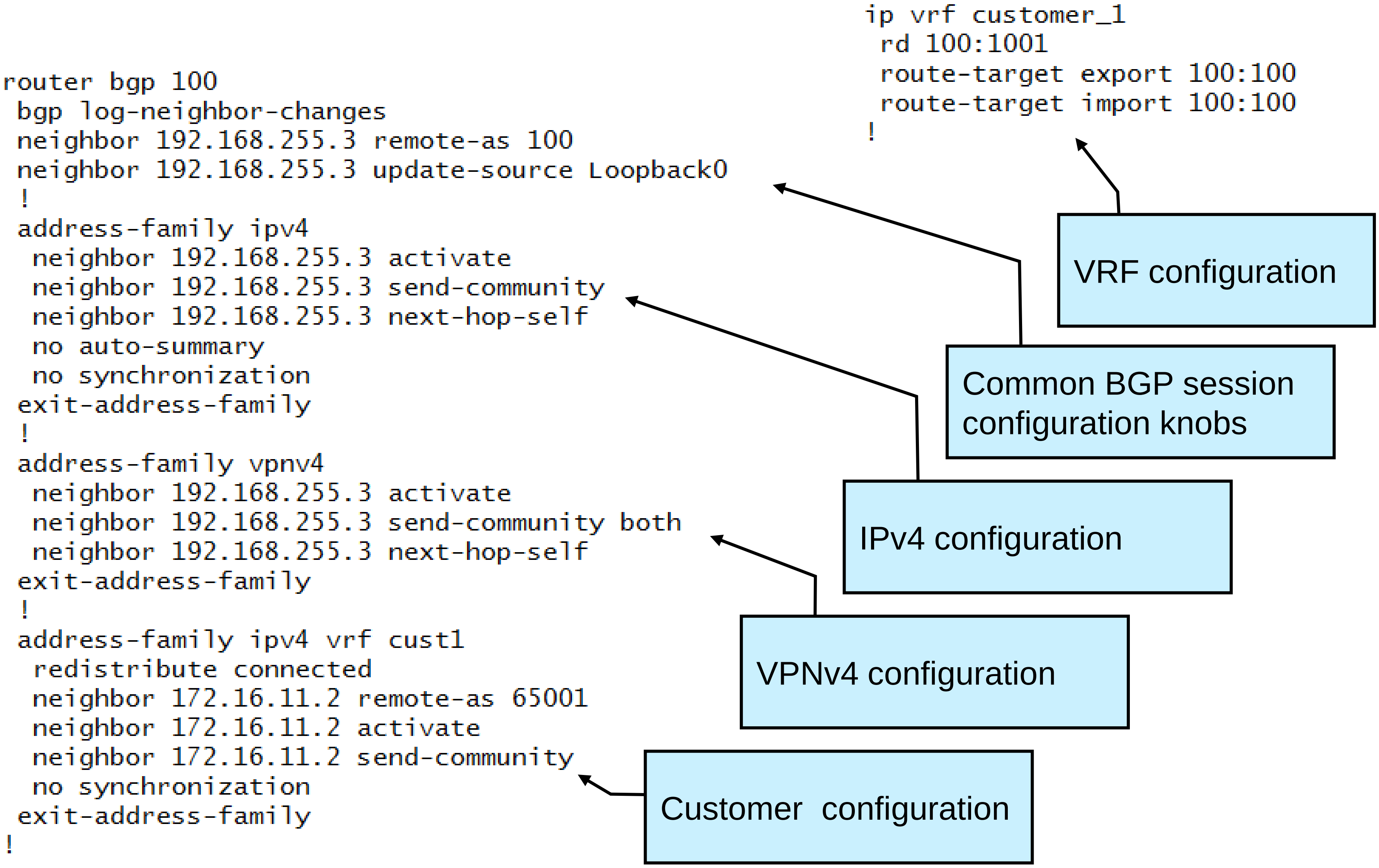
MP-BGP Update



```
⊟ Border Gateway Protocol
  ⊟ UPDATE Message
      Marker: 16 bytes
      Length: 94 bytes
      Type: UPDATE Message (2)
      Unfeasible routes length: 0 bytes
      Total path attribute length: 71 bytes
    ⊟ Path attributes
      ⊞ ORIGIN: INCOMPLETE (4 bytes)
      ⊞ AS_PATH: 65001 (7 bytes)
      ⊞ MULTI_EXIT_DISC: 0 (7 bytes)
      ⊞ LOCAL_PREF: 100 (7 bytes)
      ⊟ EXTENDED_COMMUNITIES: (11 bytes)
        ⊞ Flags: 0xc0 (Optional, Transitive, Complete)
          Type code: EXTENDED_COMMUNITIES (16)
          Length: 8 bytes
        ⊟ Carried Extended communities
            UnknownRoute Target: 100:100
      ⊟ MP_REACH_NLRI (35 bytes)
        ⊞ Flags: 0x80 (Optional, Non-transitive, Complete)
          Type code: MP_REACH_NLRI (14)
          Length: 32 bytes
          Address family: IPv4 (1)
          Subsequent address family identifier: Labeled VPN Unicast (128)
        ⊟ Next hop network address (12 bytes)
            Next hop: Empty Label Stack RD=0:0 IPv4=192.168.255.1 (12)
          Subnetwork points of attachment: 0
        ⊟ Network layer reachability information (15 bytes)
          ⊟ Label Stack=23 (bottom) RD=100:1001, IPv4=172.16.200.0/24
              MP Reach NLRI Prefix length: 112
              MP Reach NLRI Label Stack: 23 (bottom)
              MP Reach NLRI Route Distinguisher: 100:1001
              MP Reach NLRI IPv4 prefix: 172.16.200.0 (172.16.200.0)
```

```
⊟ Border Gateway Protocol
  ⊟ UPDATE Message
      Marker: 16 bytes
      Length: 55 bytes
      Type: UPDATE Message (2)
      Unfeasible routes length: 0 bytes
      Total path attribute length: 28 bytes
    ⊟ Path attributes
      ⊞ ORIGIN: INCOMPLETE (4 bytes)
      ⊞ AS_PATH: empty (3 bytes)
      ⊞ NEXT_HOP: 192.168.255.1 (7 bytes)
      ⊞ MULTI_EXIT_DISC: 0 (7 bytes)
      ⊞ LOCAL_PREF: 100 (7 bytes)
    ⊟ Network layer reachability information: 4 bytes
      ⊟ 10.200.51.0/24
          NLRI prefix length: 24
          NLRI prefix: 10.200.51.0 (10.200.51.0)
```

MP-BGP configuration



```
ip vrf customer_1
 rd 100:1001
 route-target export 100:100
 route-target import 100:100
!
```

```
router bgp 100
 bgp log-neighbor-changes
 neighbor 192.168.255.3 remote-as 100
 neighbor 192.168.255.3 update-source Loopback0
 !
 address-family ipv4
  neighbor 192.168.255.3 activate
  neighbor 192.168.255.3 send-community
  neighbor 192.168.255.3 next-hop-self
  no auto-summary
  no synchronization
 exit-address-family
 !
 address-family vpnv4
  neighbor 192.168.255.3 activate
  neighbor 192.168.255.3 send-community both
  neighbor 192.168.255.3 next-hop-self
 exit-address-family
 !
 address-family ipv4 vrf cust1
  redistribute connected
  neighbor 172.16.11.2 remote-as 65001
  neighbor 172.16.11.2 activate
  neighbor 172.16.11.2 send-community
  no synchronization
 exit-address-family
!
```

VRF configuration

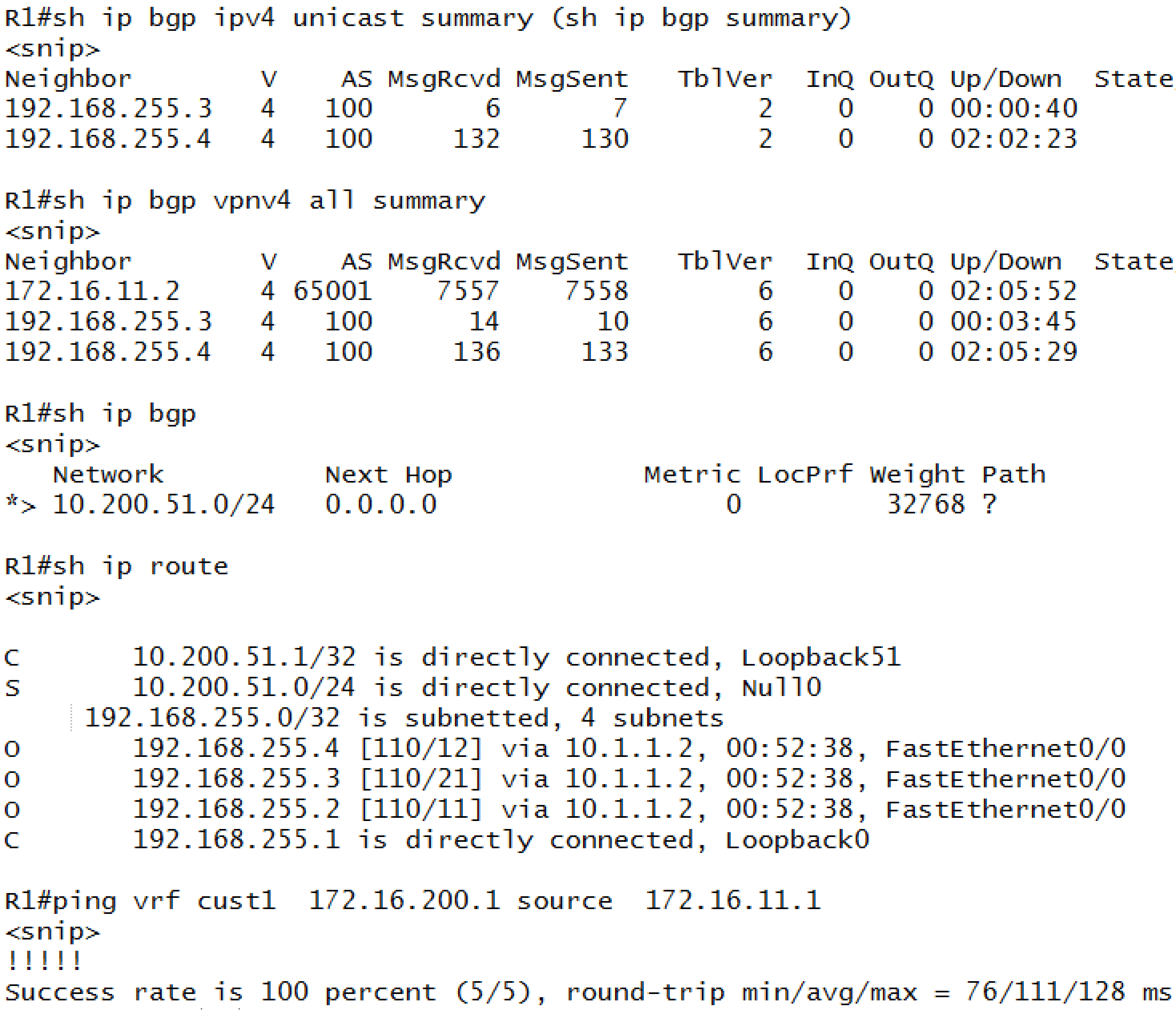
Common BGP session configuration knobs

IPv4 configuration

VPNv4 configuration

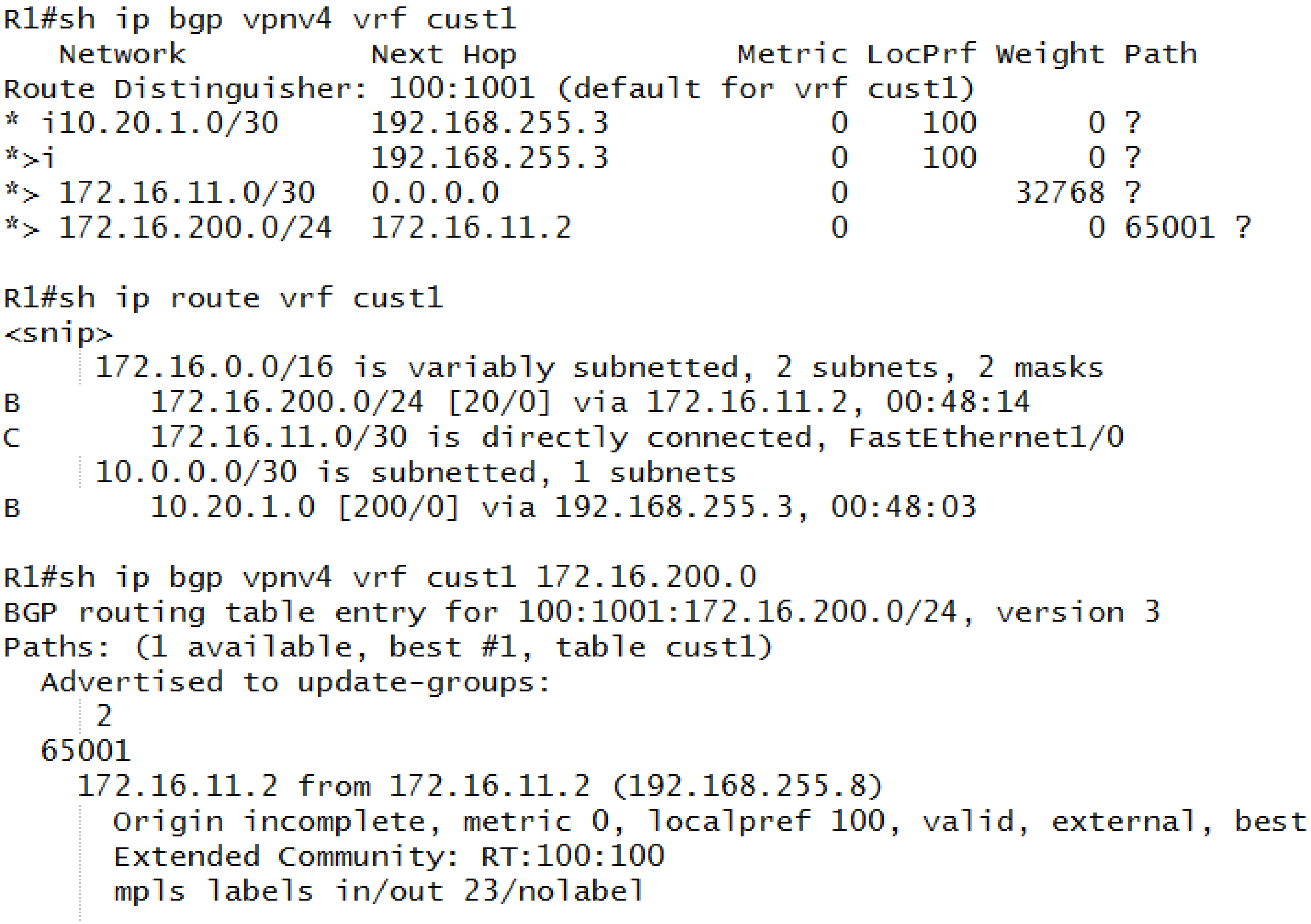
Customer configuration

MP-BGP Monitoring



```
R1#sh ip bgp ipv4 unicast summary (sh ip bgp summary)
<snip>
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
192.168.255.3   4   100       6       7        2    0    0 00:00:40        0
192.168.255.4   4   100     132     130        2    0    0 02:02:23        0

R1#sh ip bgp vpnv4 all summary
<snip>
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.11.2     4 65001    7557    7558        6    0    0 02:05:52        1
192.168.255.3   4   100      14      10        6    0    0 00:03:45        1
192.168.255.4   4   100     136     133        6    0    0 02:05:29        1

R1#sh ip bgp
<snip>
   Network          Next Hop            Metric LocPrf Weight Path
*> 10.200.51.0/24   0.0.0.0                  0         32768 ?

R1#sh ip route
<snip>

C        10.200.51.1/32 is directly connected, Loopback51
S        10.200.51.0/24 is directly connected, Null0
     192.168.255.0/32 is subnetted, 4 subnets
O        192.168.255.4 [110/12] via 10.1.1.2, 00:52:38, FastEthernet0/0
O        192.168.255.3 [110/21] via 10.1.1.2, 00:52:38, FastEthernet0/0
O        192.168.255.2 [110/11] via 10.1.1.2, 00:52:38, FastEthernet0/0
C        192.168.255.1 is directly connected, Loopback0

R1#ping vrf cust1  172.16.200.1 source  172.16.11.1
<snip>
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 76/111/128 ms
```

MP-BGP Monitoring



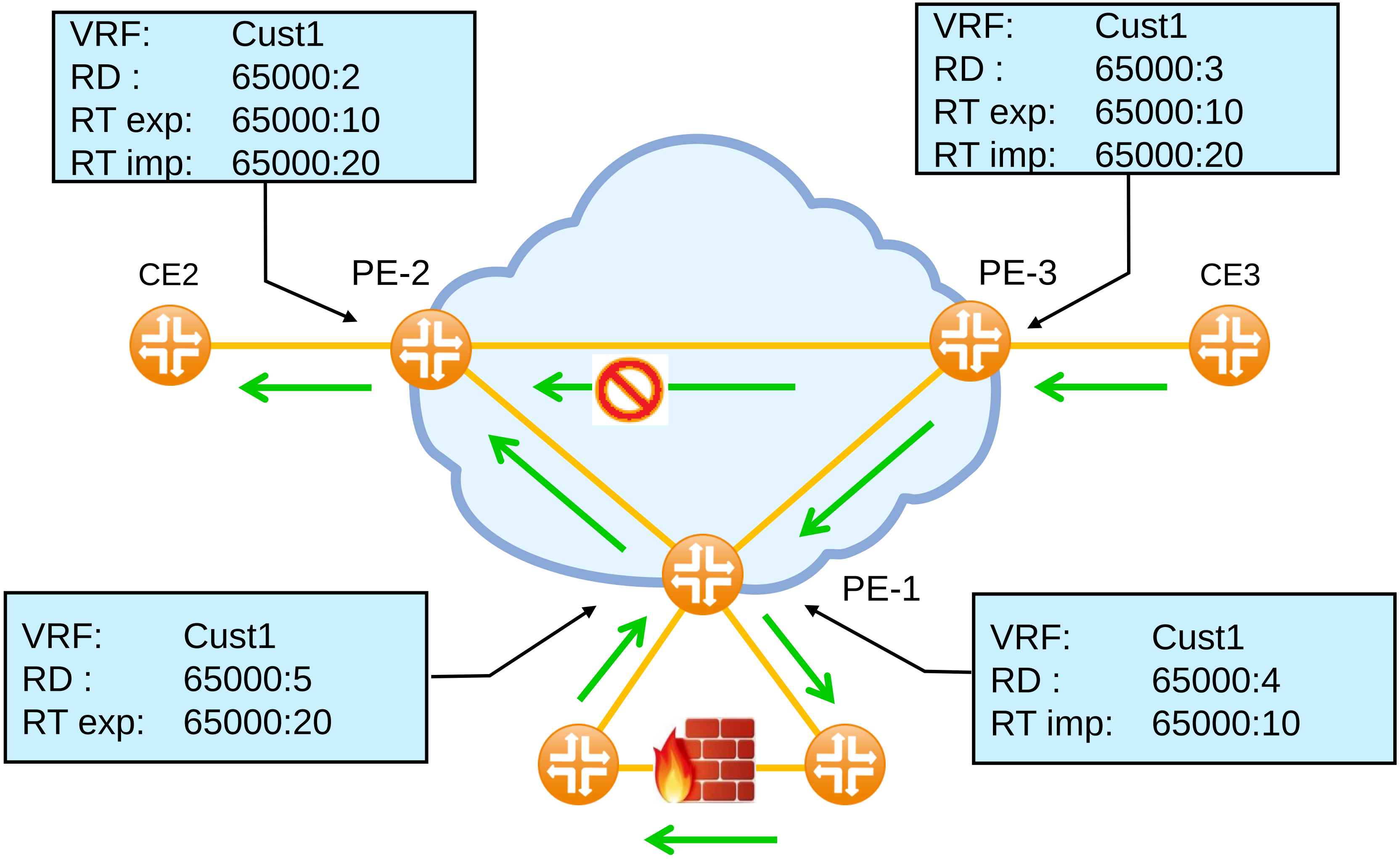
```
R1#sh ip bgp vpnv4 vrf cust1
   Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 100:1001 (default for vrf cust1)
* i10.20.1.0/30     192.168.255.3            0    100      0 ?
*>i                 192.168.255.3            0    100      0 ?
*> 172.16.11.0/30   0.0.0.0                  0         32768 ?
*> 172.16.200.0/24  172.16.11.2              0             0 65001 ?

R1#sh ip route vrf cust1
<snip>
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
B       172.16.200.0/24 [20/0] via 172.16.11.2, 00:48:14
C       172.16.11.0/30 is directly connected, FastEthernet1/0
     10.0.0.0/30 is subnetted, 1 subnets
B       10.20.1.0 [200/0] via 192.168.255.3, 00:48:03

R1#sh ip bgp vpnv4 vrf cust1 172.16.200.0
BGP routing table entry for 100:1001:172.16.200.0/24, version 3
Paths: (1 available, best #1, table cust1)
  Advertised to update-groups:
     2
  65001
    172.16.11.2 from 172.16.11.2 (192.168.255.8)
      Origin incomplete, metric 0, localpref 100, valid, external, best
      Extended Community: RT:100:100
      mpls labels in/out 23/nolabel
```

VPN Topologies

- RTs are used to decide which routes are visible in a VRF
- By exporting routes with selective RTs and by selectively importing routes, different VPN models can be built

- Any-to-any VPN
  - All the sites of a VPN can communicate directly with each other. All sites export the routes using the same RT and import routes that have the same export RT
- Hub and Spoke VPN
  - w/o connectivity between Spokes or w connectivity only via Hub)
  - This VPN model is typically used when a VPN has a central site (Hub) and regional sites (spokes). The central site provides services to the regional sites and there is no or optional need for the regional sites to communicate with each other directly
- Inter-VPN
  - created by exporting or importing one or more addresses in a VRF with multiple RTs
  - Common example of an inter-VPN service is the management VPN, managed CE part of the customer and mngmt VPN

Hub and Spoke model



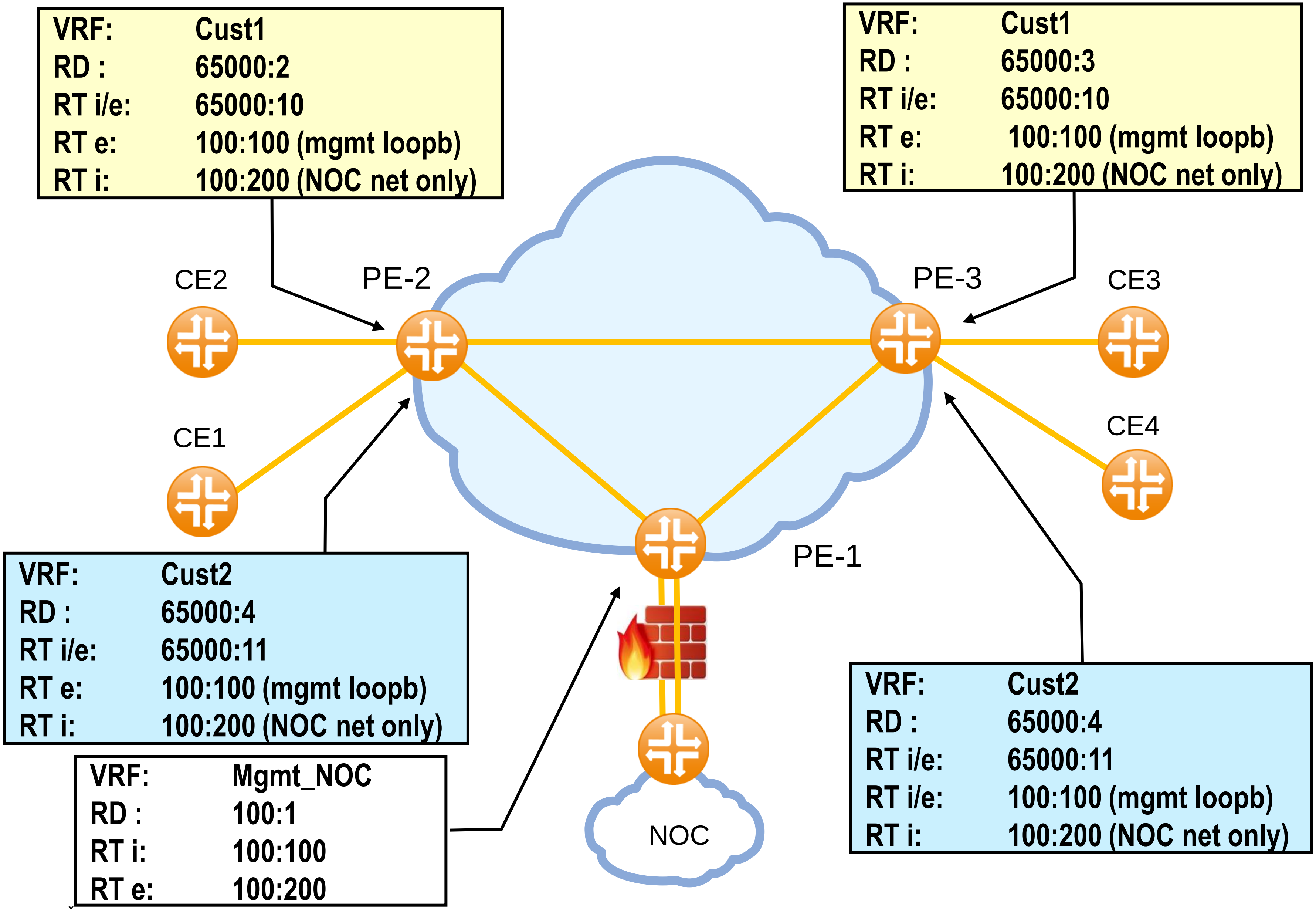
VRF:	Cust1
RD :	65000:2
RT exp:	65000:10
RT imp:	65000:20

VRF:	Cust1
RD :	65000:3
RT exp:	65000:10
RT imp:	65000:20

VRF:	Cust1
RD :	65000:5
RT exp:	65000:20

VRF:	Cust1
RD :	65000:4
RT imp:	65000:10

Inter-VPN model, managed CEs



VRF:	Cust1
RD :	65000:2
RT i/e:	65000:10
RT e:	100:100 (mgmt loopb)
RT i:	100:200 (NOC net only)

VRF:	Cust1
RD :	65000:3
RT i/e:	65000:10
RT e:	100:100 (mgmt loopb)
RT i:	100:200 (NOC net only)

VRF:	Cust2
RD :	65000:4
RT i/e:	65000:11
RT e:	100:100 (mgmt loopb)
RT i:	100:200 (NOC net only)

VRF:	Mgmt_NOC
RD :	100:1
RT i:	100:100
RT e:	100:200

VRF:	Cust2
RD :	65000:4
RT i/e:	65000:11
RT i/e:	100:100 (mgmt loopb)
RT i:	100:200 (NOC net only)

AS-override, allowas-in, SOO

- The same AS number can be used for all the sites of a VPN to conserve the number of private AS numbers.

- AS-override
  - replace the AS number of originating router with the ASN of the sending BGP router but only if it is the same as the AS of the neighbor
  - usually used on the PE in egress direction towards CE

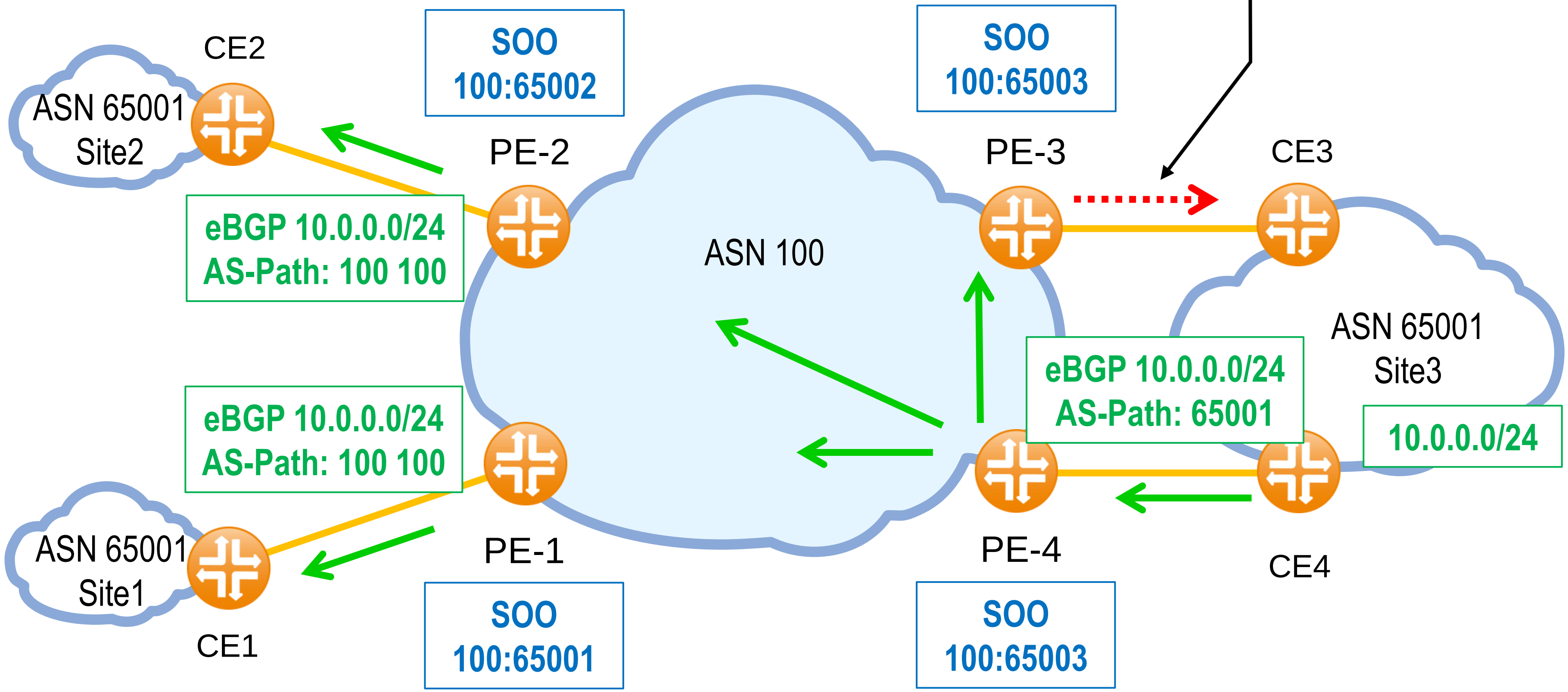
- allowas-in
  - allows the loop prevention to be ignored
  - usually used on the CE to inspect incoming BGP updates

- Site of Origin
  - SOO can be used to avoid an AS-override/allowas-in induced route loop. SOO is an extended community attribute attached to a BGP route used to identify the origin of the route.
  - If the attached SOO is equal to the configured SOO for a BGP peering, the route is blocked from being advertised
  - Extended community, configured via route maps for incoming routes

SOO and AS_override

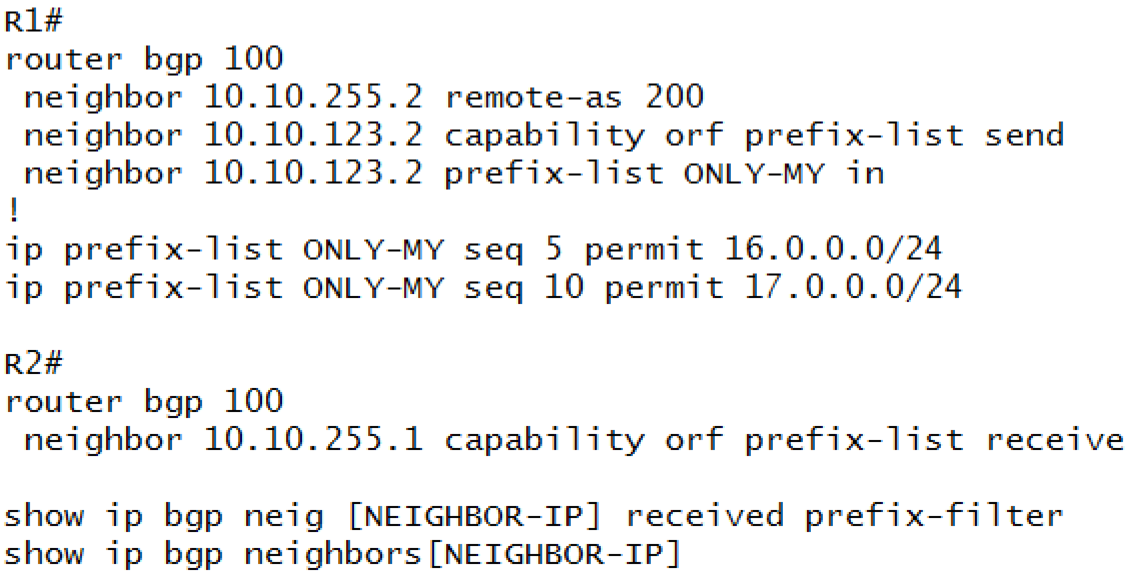
Update removed by CE3 in case of AS_override only

Update not generated by PE-3 in case of SOO configured



ORF - Outbound Route Filtering

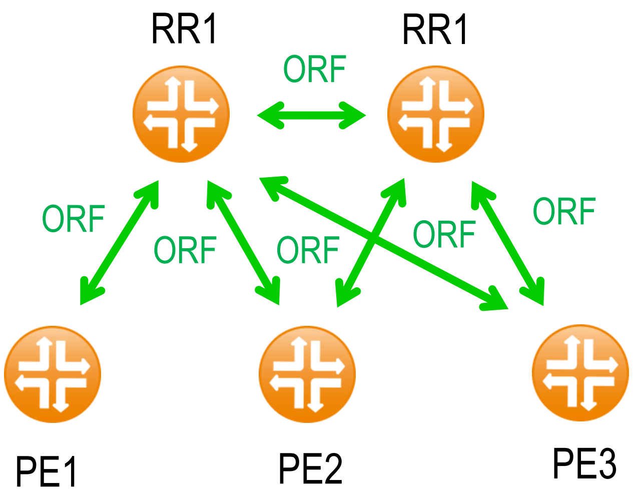
- iBGP full mesh between PEs (with or without RRs) results in flooding all VPNs routes to all PEs
- Therefore each PE will discard any VPN-IPv4 route that hasn’t a route-target configured to be imported in any of the attached VRFs
  - This reduces the amount of information each PE has to store
- ORF allows a router to tell its neighbors which filter (RT/prefix based) to use prior to propagate BGP updates
  - Extended Communities ORF type used



```
R1#
router bgp 100
 neighbor 10.10.255.2 remote-as 200
 neighbor 10.10.123.2 capability orf prefix-list send
 neighbor 10.10.123.2 prefix-list ONLY-MY in
!
ip prefix-list ONLY-MY seq 5 permit 16.0.0.0/24
ip prefix-list ONLY-MY seq 10 permit 17.0.0.0/24

R2#
router bgp 100
 neighbor 10.10.255.1 capability orf prefix-list receive

show ip bgp neig [NEIGHBOR-IP] received prefix-filter
show ip bgp neighbors[NEIGHBOR-IP]
```

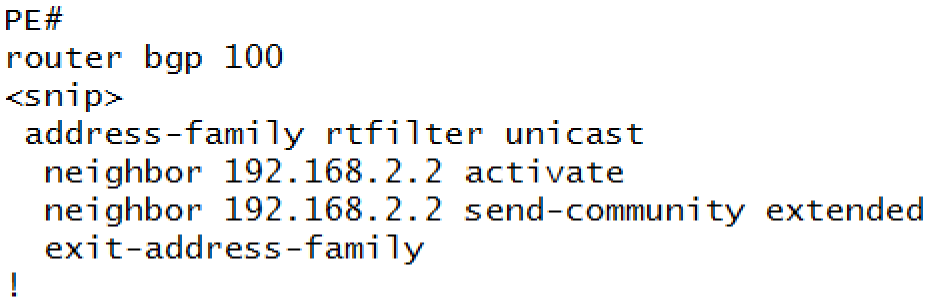
Use of Route Reflectors

- Scalability of VPN route distribution can be increased by use of BGP Route Reflectors (RR)
- By default no ORF – less peers
- Two ways to partition VPN IPv4 routes among different RRs
  - Each RR is pre-configured with a list of RTs/prefixes or RT Constrained Route Distribution feature is used
  - Each PE is a client of a subset of RRs, no ORF from clients but ORF between RRs



ORFs based the list of RTs

The feature allows the PE to propagate RT membership towards RR and use the RT membership to limit the VPN routing information maintained at the PE and RR. RR sends to PE only relevant ones.



```
PE#
router bgp 100
<snip>
 address-family rtfilter unicast
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 send-community extended
  exit-address-family
!
```

Constrained Route Distribution

Security of BGP/MPLS VPNs

- Built-in security features
  - Access to VPNs is tightly controlled by the PEs
  - Total address separation by use of VPN IPv4 addresses
  - Separation of routing information by use of route targets

- Vulnerabilities
  - Misconfiguration of the core and attacks within the core
  - Security of the access network

- Additional Security can be provided by combining IPSec and MPLS
  - End-to-end IPSec overlaid on an MPLS VPN

Ďakujem za pozornosť

roman dot kaloc at gmail dot com