MP-BGP/MPLS VPN

Tawfiq Khan
TCOM 610
George Mason University

MP-BGP (RFC4760)

- Why
  - Current BGP-4 is used to advertise IPv4 address reachability only
  - ISP use BGP extensively.
  - It will be easier to extends BGP for multiple network layer protocols (IPv6, multicast, MPLS-VPN, CLNS etc) than to create new protocol
- How
  - Introduce new BGP “capability advertisement” in open message to determine if a peer supports MP-BGP extensions; Exchange of MP-NRLI capability must be exchanged at session set-up.
  - Introduce optional non-transitive attributes
    - MP_REACH_NLRI -- type 14
    - MP_UNREACH_NLRI -- type 15

New Attributes

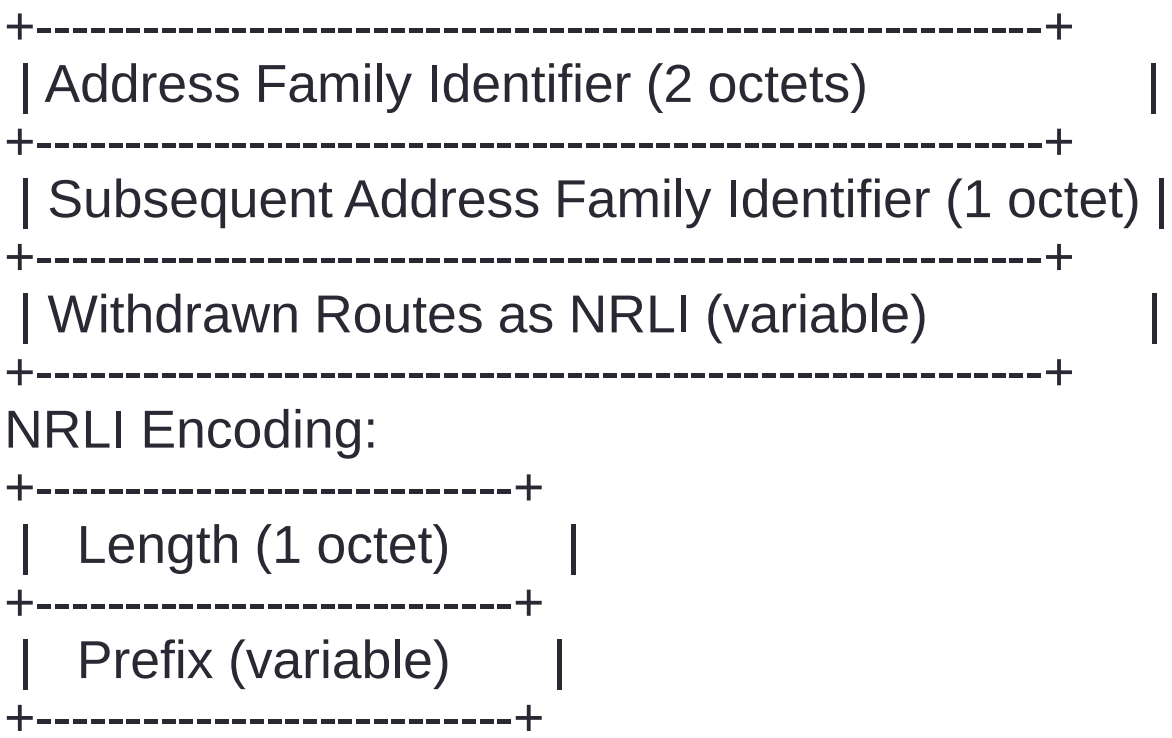
- One or more of triples: (AFI, SubAFI, NRLI)
- MP_REACH_NLRI is used to
  - advertise a feasible route to a peer
  - advertise the NL address as the next hop to the destinations listed in the NLRI
  - report the Subnetwork Points of Attachment (SNPAs) within the local system
- MP_UNREACH_NLRI is used to
  - withdraw multiple unfeasible routes

MP-BGP Reachability Attributes

AFI (2 octets)	Sub AFI (1 octet)
Len of Next Hop NetAddress (1 octet)	Next Hop NetAddress (variable)
Number of SNPAs (1 octet)	
Len of 1st NPA(1 octet)	First SNPA (variable)
…….	
Length of Last SNPA (1 octet)	Last SNPA (variable)
NRLI (variable)	

AFI: IP=1, IPv6=2, IPX=11
Sub AFI: Unicast=1, Multicast=2, Uni/Multi=3, Label=4, VPN=128

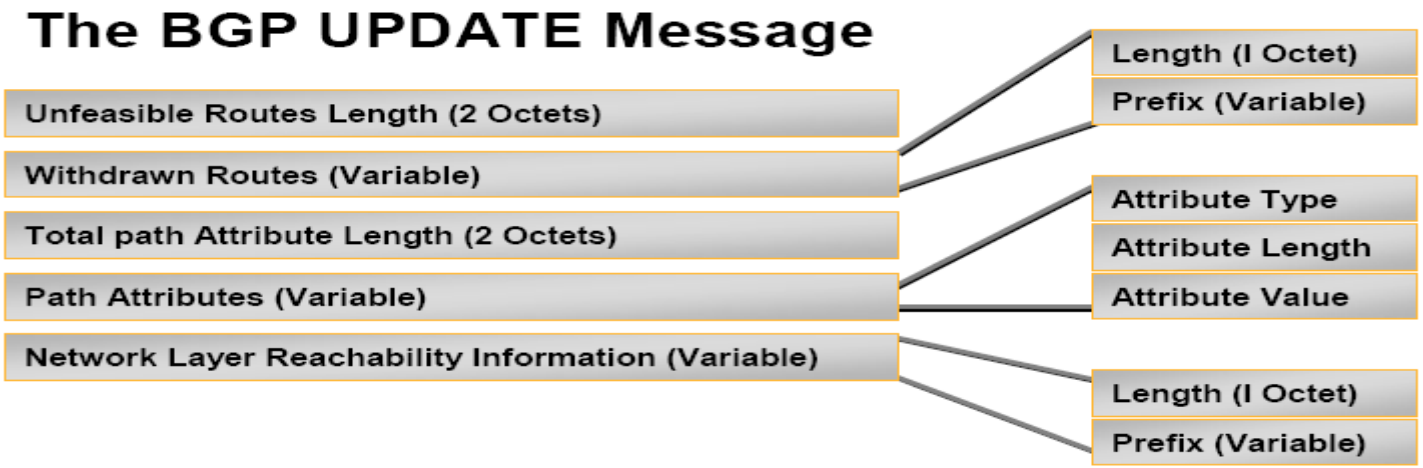
MP-BGP Unreachability Attributes



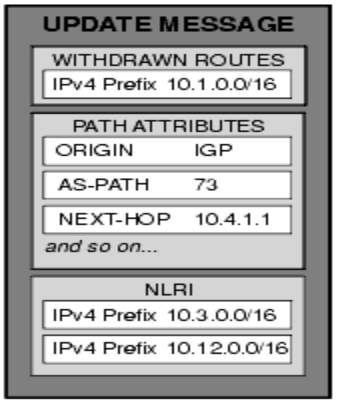
NRLI Encoding

- Length:
  - length in bits of the address prefix. A length of zero indicates a prefix that matches all addresses
- Prefix:
  - address prefix followed by enough trailing bits to make the end of the field fall on an octet boundary. Note that the value of trailing bits is irrelevant
  - Example: 210.8.0.0/23: len=23, prefix=210.8

MP-BGP Update

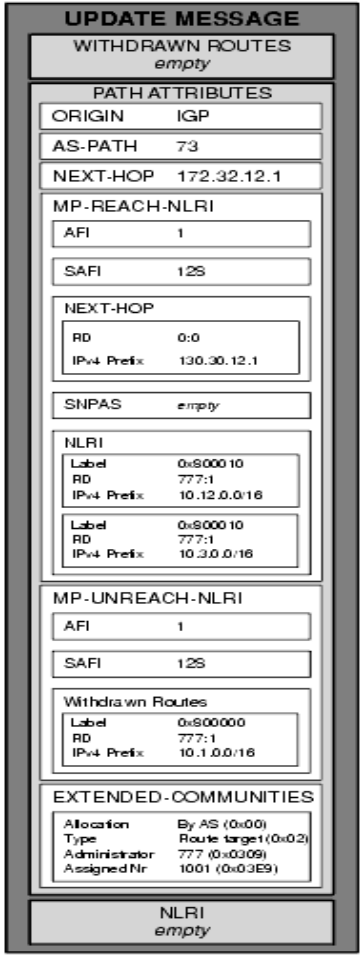


MP-BGP Update



Standard Update Message

Extended Update Message



MPLS/BGP VPN (RFC4363)

- MPLS/VPN basic concepts
  - What it is?
  - Why we need it? – Lower cost and Scalability
  - How it works? – MP-BGP, VRF, MPLS etc.

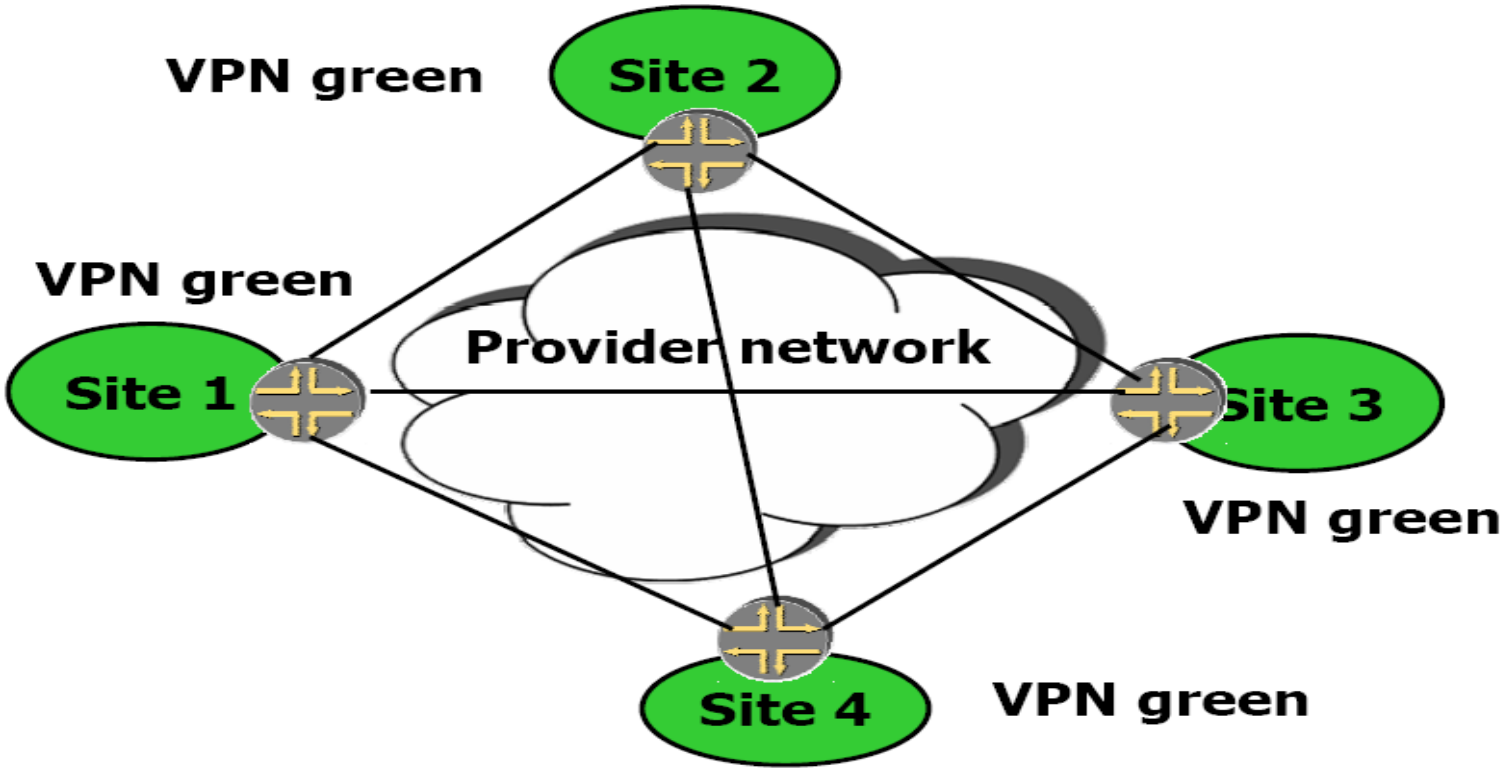
VPN: introduction

- Virtual Private Networks
  - Interconnect geographically separate customer sites – connectivity.
  - Over a shared Service Provider Network infrastructure – lower cost.
  - With the same privacy and guarantees as a private network – private.
- MPLS/BGP VPN
  - Peer Model
  - MP-BGP as customer route distribution protocol
  - MPLS as forwarding plane

Overlay Models

- Traditional CE based VPN.
- Sites are connected with p2p links – leased lines, FR circuits, ATM circuits, GRE, IPsec.
- CE peer with other CEs at Layer 3.
- The SP needs to design and operate “virtual backbones” for all the customers – scaling issue.
- Problem with VPNs that have a large number of sites (full mesh?).
- Adding a new site requires configuring all the existing sites.

Overlay Model (continue)



Peer Model

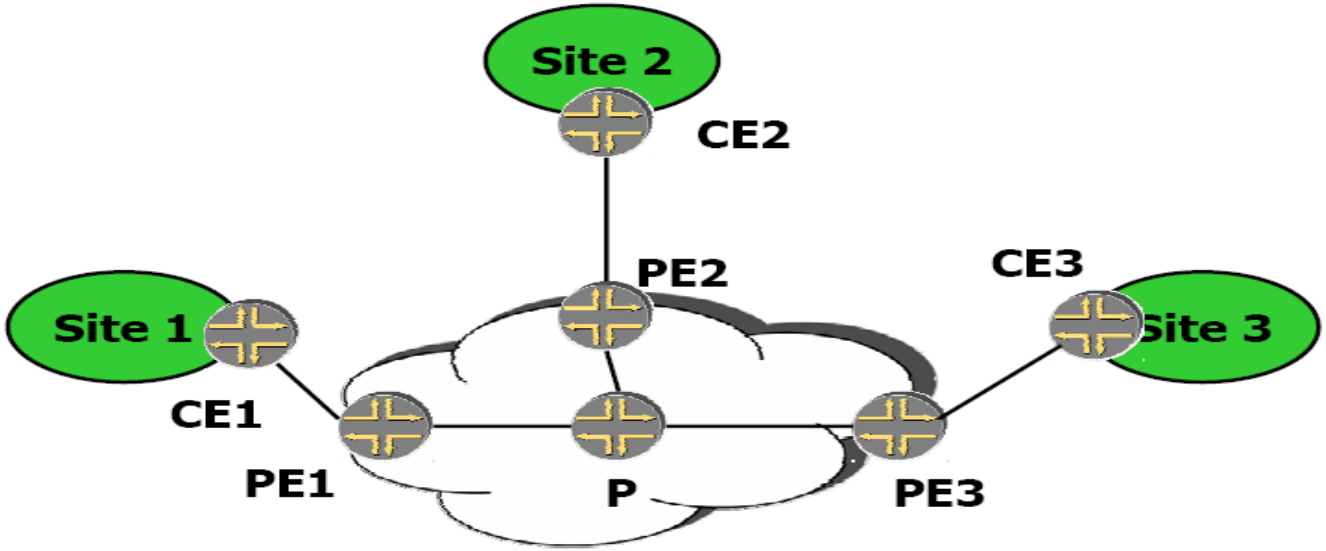
Goal: Solve the scaling issues.

- Support thousands of VPNs
- Support VPNs with hundreds of sites per VPN
- Support overlapping address space.

Peer Model (continue)

- CE router peers with a PE router, not with other CE routers.
- Adding/deleting a new site requires only configuring the PE router connected to the site.
- A PE router only needs to maintain routes for the VPNs whose sites are directly connected.
- Any-to-Any in nature.
- BGP/MPLS VPN belongs to Peer model

Peer Model (continue)



BGP-MPLS VPN: How it works

- Defined in RFC4364, RFC4577, RFC4684
  - Constrained Distribution of routing information
    - Sites connectivity control by BGP policies
  - Separation of forwarding (VRF)
    - Multiple forwarding tables
  - Address Family extension
    - To allow overlapping address space in VPN
  - Forwarding with MPLS
    - Decouple forwarding (label) with IP header info.

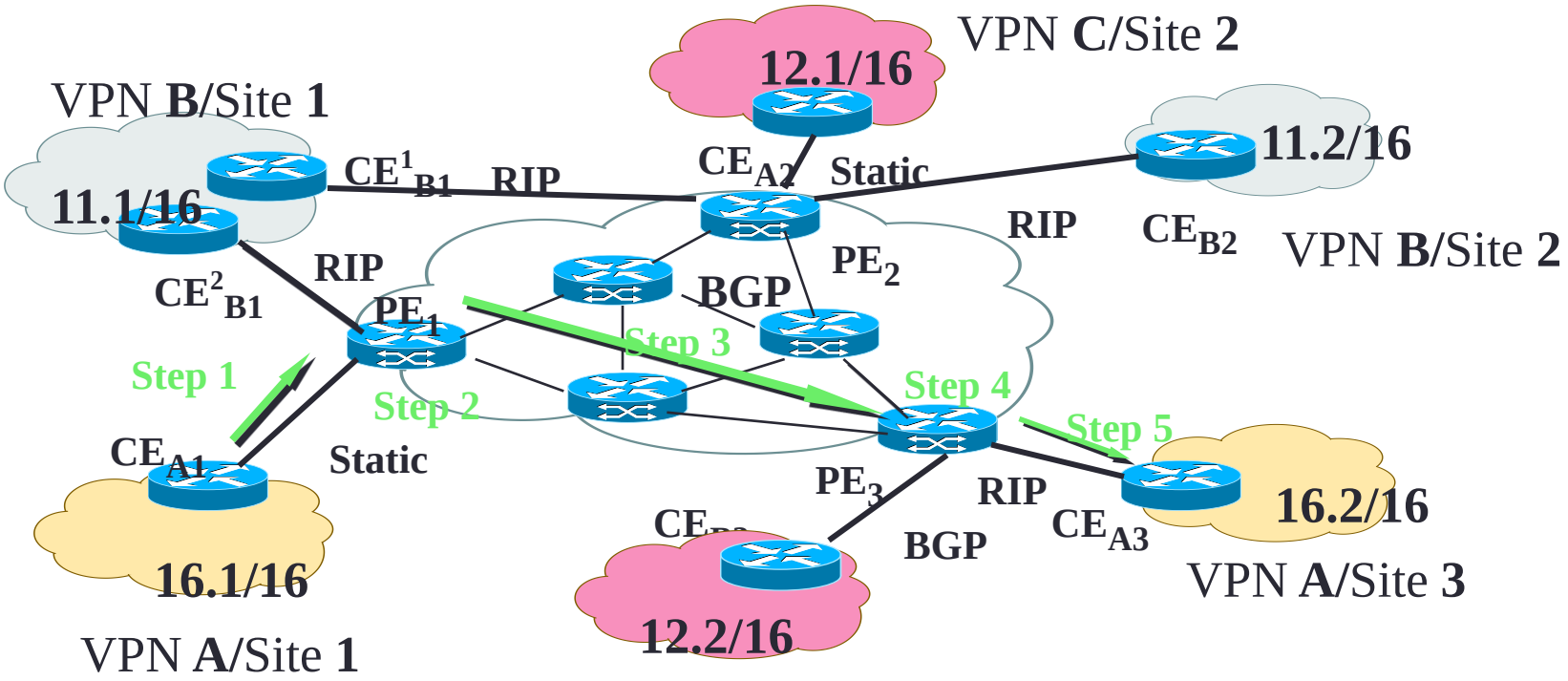
Constrained Routes Distribution

- Step 1: from site (CE) to service provider (PE)
  - e.g., via RIP, or static routing, or BGP, or OSPF
- Step 2: export to provider's BGP at ingress PE
- Step 3: within/across service provider(s) (among PEs):
  - via MP-BGP
- Step 4: import from provider's BGP at egress PE
- Step 5: from service provider (PE) to site (CE)
  - e.g., via RIP, or static routing, or BGP, or OSPF

Constrained Routes Distribution

- Constrained Distribution of Routing Information Occurs during Steps 2, 3, 4
- Performed by Service Provider using route filtering based on BGP Extended Community attribute
  - BGP Community is attached by ingress PE at Step 2
  - route filtering based on BGP Community is performed by egress PE at Step 4

Constrained Routes Distribution



Routes Target (RT)

```
!
ip vrf red
rd 1:1
route-target export 1:1
route-target import 1:1
```

- To control policy about who sees what routes
- 64-bit quantity (2 bytes type, 6 bytes value)
- Carried as an extended BGP community in MP-BGP routes
- Typically written as ASN:Index
- Each VRF ‘imports’ and ‘exports’ one or more RTs
  - Exported RTs are carried in VPNv4 BGP
  - Imported RTs are local to the PE
- A PE that imports an RT installs that route in its routing table

Route Distinguisher (RD)

- 64 Bit RD: 16 bit Type, 48 bits Value
  - Administrated by SP and unique within PE
  - No specific syntax meaning besides making VPN routes unique
  - Used to be assigned as the same as RT, but RD and RT really serve totally different purpose
- VPN Route: 64 bit RD, followed by 32 bit IPv4 address
  - Make total VPN route: 96 bits
  - Guarantee unique within SP
- VPN routes are attached with RT to control its import/export into VPNs

Separation of forwarding

- Problem:
  - How to constrain distribution of routing information at PE that has sites from multiple (disjoint) VPNs attached to it ?
  - Single Forwarding Table on PE doesn’t allow per VPN segregation of routing information
- Solution:
  - PE maintains multiple Forwarding Tables
    - one per set of directly attached sites with common VPN membership
    - e.g., one for all the directly attached sites that are in just one particular VPN
  - Enables (in conjunction with route filtering) per VPN segregation of routing information on PE

Separation of forwarding

- Each Forwarding Table is populated from:
  - routes received from directly connected CE(s) of the site(s) associated with the Forwarding Table
  - routes receives from other PEs (via BGP)
    - restricted to only the routes of the VPN(s) the site(s) is in via route filtering based on BGP RT Attribute
- Each customer port on PE is associated with a particular Forwarding Table.
- Provides PE with per site (per VPN) forwarding information for packets received from CEs

- VRF: VPN routing and forwarding

Address Family Extension

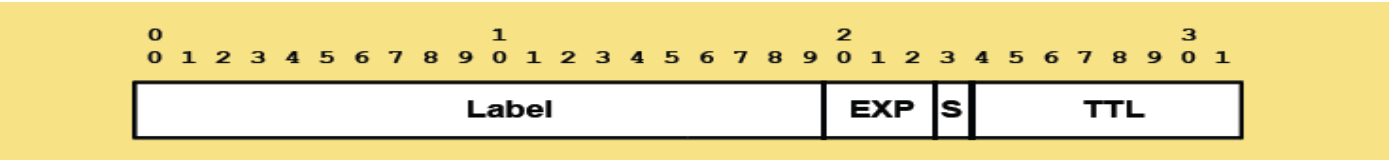
- Problem:
  - To allow each VPN to use overlapping address space
- Solution:
  - Add an unique ID in front of prefixes to make them unique.
- Result:
  - SAFI 128 (VPNv4) routes:
  - Constructed by concatenating an IP address and an 8-byte unique identifier called the route distinguisher (RT).
  - Route Distinguisher doesn’t have to be the same for all routes in the VPN
  - Typical values: AS:number / IPaddress:number

What is MPLS

- Multi Protocol Label Switching
- MPLS is an efficient encapsulation mechanism
- Uses labels appended to packets for transport data
- MPLS packets can run on other Layer 2 technologies such as ATM, FR, PPP, POS, Ethernet
- Other Layer 2 technologies can be run over an MPLS network
- Labels can be used as designators
  - Forwarding Equivalence Class (FEC)
  - For example — IP prefixes, ATM VC, or a bandwidth guaranteed path

MPLS (continue)

- Header



  - Labels(20bits):Exp(3bits):Stack(1bit):TTL(8bits)
- Forwarding Plane
  - Bind FEC to a label (LFIB)
- Control Plane
  - Label distribution: LDP or RSVP

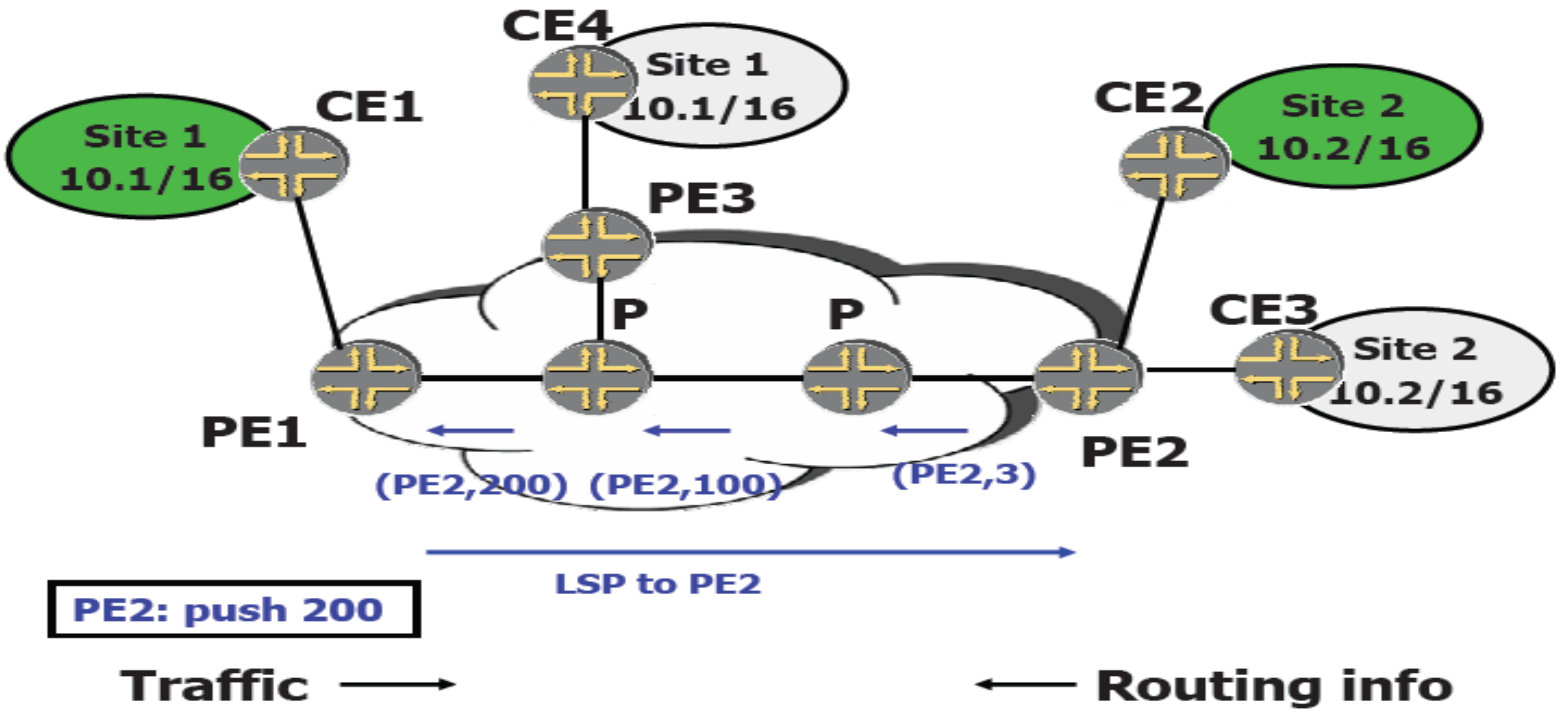
Why MPLS?

- VPN-IP addresses (SAFI 128) are used by the routing protocols, but do not appear in headers of IP packets.
- Need a way to forward traffic along routes to VPN-IP addresses.
- MPLS decouples forwarding from the destination information.

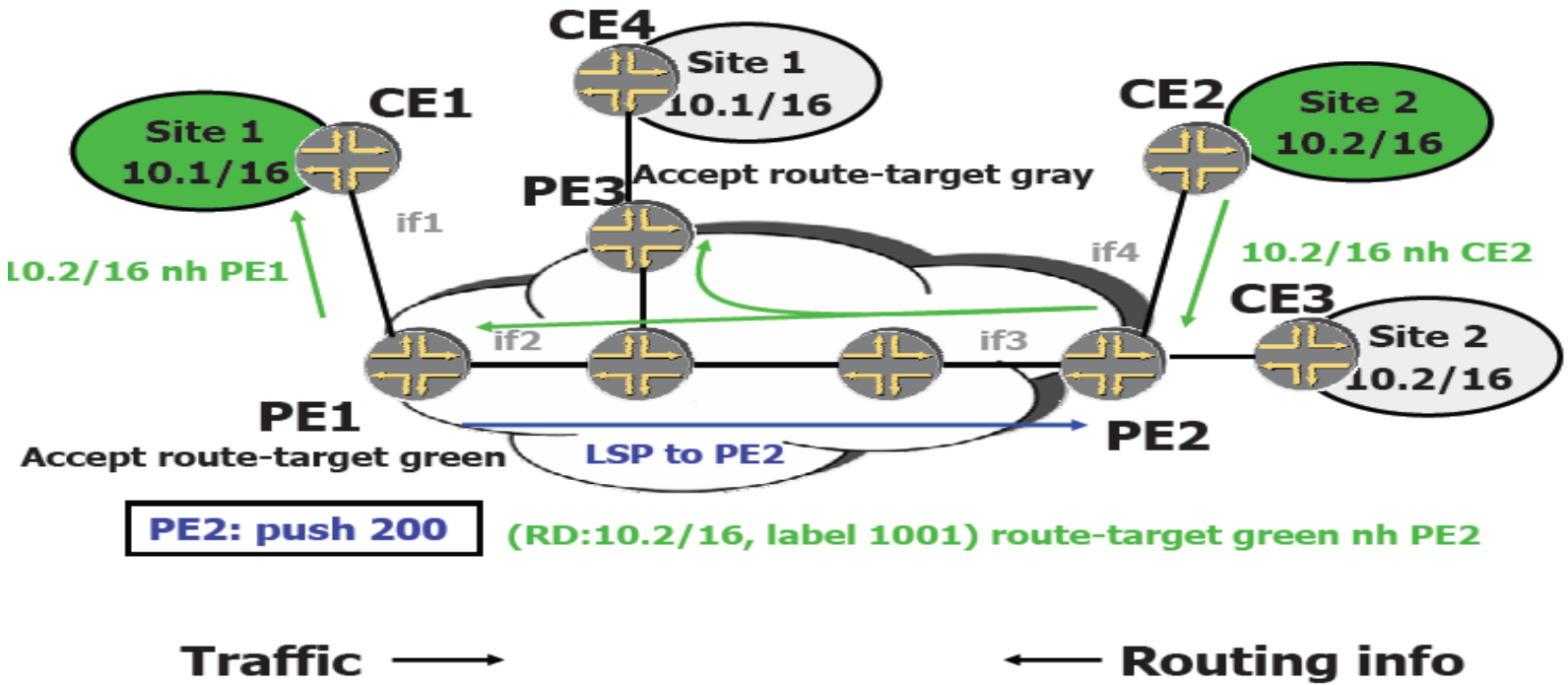
Forwarding with MPLS

- The idea:
  - Use a label to reach remote PE (BGP next-hop). Also called LSP label.
  - Use a second label to identify VPN interface at the remote PE. Also called VPN label.
- The LSP label is distributed by either LDP or RSVP.
- The VPN label is distributed by BGP, along with the VPN-IP address.
- Traffic will carry both labels, the LSP label and the VPN label.
- The remote PE makes the forwarding decision based on the VPN label.

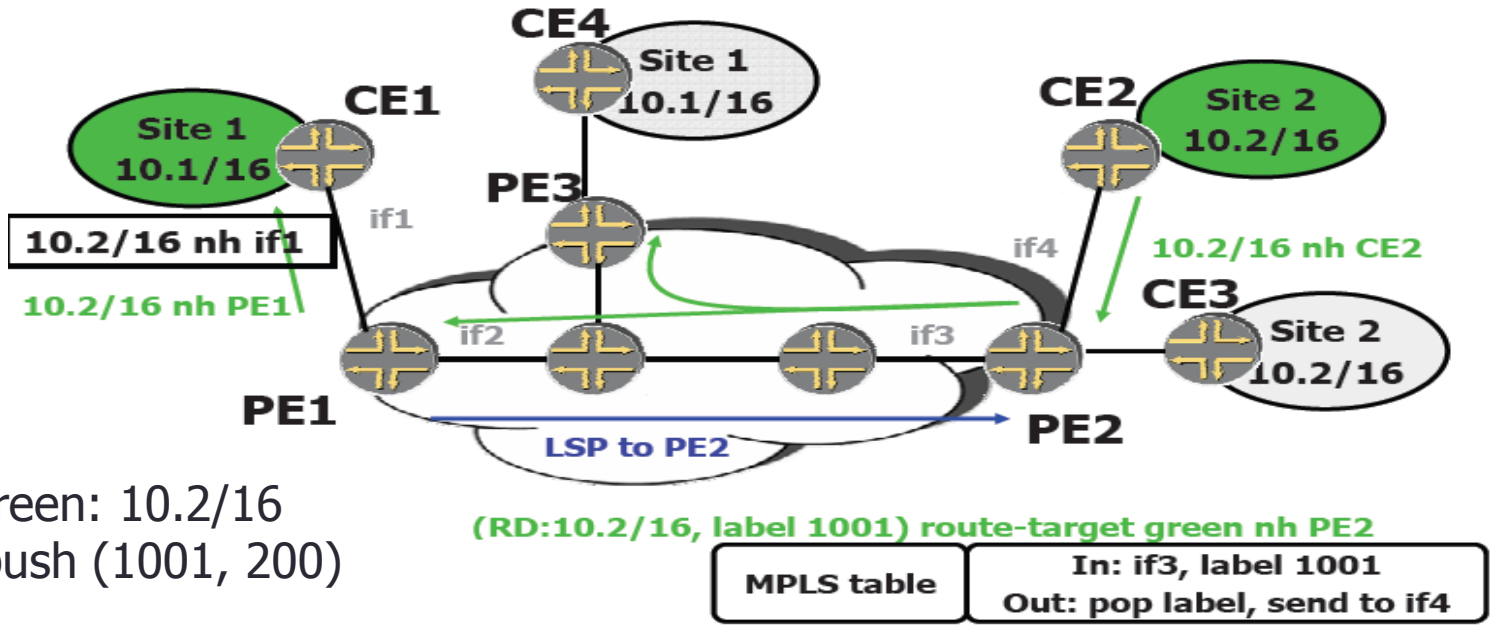
Step 1: LSP setup



Step 2: Route distribution

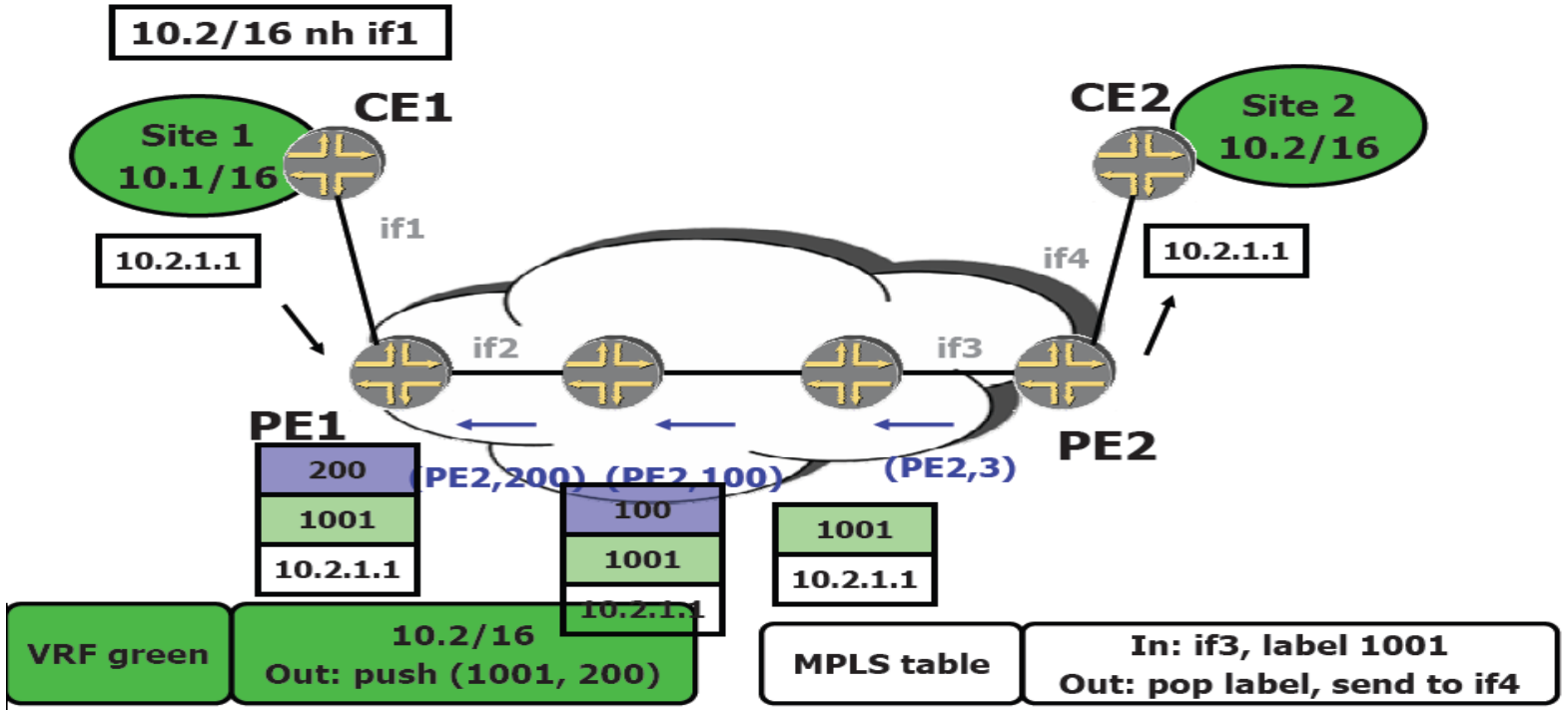


Step 3: forwarding tables



VRF green: 10.2/16
Out: push (1001, 200)

Step 4: forwarding traffic



Steps Summary

- Full mesh of MP-BGP between all PEs.
  - Control plane Delivery VPN routes plus labels
- MPLS connectivity between all PEs.
  - Forwarding plane LSP Reachabilty between PEs
- BGP advertises a label along with the VPN-IP address. This determines the next-hop to use when receiving traffic.

MPLS VPN Summary

- Control plane:
  - Use VRF table inside PE to separate different VPN routes
  - Use LDP/RSVP to distribute LSP label for PE reachability
  - Use MP-BGP to distribute VPN routes and VPN labels between remote PEs
    - Use SAFI 128 address family
    - RD is used to uniquely identify VPN routes in the SP core.
    - RT is used to import/export routes from/to VRF table.

MPLS VPN Summary (cont)

- Forwarding plane: dual stack
  - Outer label (MPLS LSP Label) :
    - Forward traffic to remote PE (MP-BGP next hop) across nodes that don’t have routing information for the packet’s final destination
  - Inner Label (VPN Label):
    - To carry VPN traffic to its outgoing VPN interface on the remote PE.

Scaling properties

- For a CE, only one routing peering (CE-PE) is needed, regardless of the number of sites in the VPN.
- Adding a new site requires configuration of one PE regardless of the number of sites (constant # of changes required to add a new site)
- PE has to maintain routes only for the VPNs to which it is connected.
- P routers don’t have to maintain VPN routes at all.
- Can use overlapping address spaces – efficient use of private IP addresses.
- Route distinguishers are structured so that each service provider can manage its own number space.