

VPLS met technische opmerking voor BGP-signalering

Inhoud

[Inleiding](#)

[Probleem](#)

[Architectuur van de oplossing](#)

[Identificatie van AE-routerendpoints](#)

[VPLS-identiteit- en MPLS-labels](#)

[Insluitingsinformatie](#)

[VPLS BGP-voorvoegsel voor automatische detectie en VPLS BGP-signaleringsvoorvoegsel](#)

[Cisco IOS-softwareconfiguratie](#)

[geadvertiseerd labelblok](#)

[Routeonderscheiding en routedoelstellingen](#)

[Voorbeelden van configuratie van PE1](#)

[Label bereik controleren](#)

[Etikels controleren](#)

[Label Block controleren](#)

[Geavanceerd prefixeren controleren](#)

[Voorvoegsel in detail bekijken](#)

[Advers, Ontvang en proces van labelblokken in BGP-update-berichten](#)

[PE2: Ontvang BGP-update](#)

[PE2: Een label zoeken](#)

[PE2: Prefixeren bij PE1 verzenden](#)

[PE1: Een label zoeken](#)

[Aanvullende verificatieopdrachten](#)

[Meervoudige L2VPN-VPLS prefixes geadverteerd door PE-router voor één VFI](#)

[PE1-configuratie](#)

[PE2-configuratie](#)

[Oorspronkelijke labelblokken](#)

[PE1 en PE2 uitwisseling](#)

[Analyse van PE1 en PE2-uitwisseling](#)

[Prefixdetails](#)

[Interoperabiliteit](#)

Inleiding

Dit document beschrijft op BGP (Border Gateway Protocol) gebaseerde automatische ontdekking voor een Virtual Private LAN Service (VPLS) met BGP-signalering. Auto-discovery is een middel

voor een provider edge (PE) om te leren welke op afstand geplaatste PE's deel uitmaken van een bepaald VPLS-domein. Signaling is een middel voor een PE om het pseudodraadlabel te leren dat door een gegeven afstandsPE voor een bepaald VPLS-domein wordt verwacht.

Raadpleeg de documenten van de Internet Engineering Task Force:

- [RFC 4762 Virtual Private LAN Service \(VPLS\) met Label Distribution Protocol \(LDP\)-signalering](#) beschrijft BGP-automatische ontdekking met LDP-signalering (Label Distribution Protocol) voor VPLS (ook bekend als Martini).
- [RFC 4761 Virtual Private LAN Service \(VPLS\) met BGP voor automatische detectie en signaling](#) beschrijft BGP-automatische detectie en BGP-signalering voor VPLS (ook bekend als Kompella).

Dit document is gericht op RFC 4761. Met RFC 4761 houdt de BGP Network Layer Reachability Information (NLRI) van de BGP-updates de informatie voor zowel automatische detectie als signaling in. Wanneer externe PE-routers deze BGP-update ontvangen, beschikken ze over alle informatie die nodig is om een volledig netwerk van pseudodraden voor VPLS op te zetten. BGP-automatische detectie en BGP-signalering gebruiken dezelfde BGP-adresfamilie.

De opdrachtregel interface (CLI) en de uitvoer komen van Cisco IOS[®] Software. De configuratie en functionaliteit zijn zeer vergelijkbaar in Cisco IOS-XR software en Cisco NX-OS software.

Probleem

VPLS bestaat uit een reeks pseudodraden (PW) op een point-to-multipoint manier. Tot nu toe werd LDP gebruikt om de pseudodraden tussen de PE-routers te signaleren. Dus werd er een gerichte LDP-sessie gemarkeerd die labels te gebruiken waarvoor pseudo-draad tussen één PE-routers bestond. U kunt de reeks PE-routers die aan één VPLS-domein hebben deelgenomen handmatig configureren of u kunt BGP gebruiken om de configuratie automatisch te ontdekken. Om deze automatische ontdekking uit te voeren, maakte BGP bekend welke PE er lid van was, waarvan VPLS-domein. Maar zelfs met de automatische detectie van BGP werd LDP gebruikt om het MPLS-label (Multiprotocol Label Switching)-labels (VC) en pseudobedraging-ID te signaleren.

Het is nu mogelijk om BGP te gebruiken om de pseudodraden tussen de PE routers te signaleren.

Wanneer een pseudodraad tussen een paar routers moet worden ingesteld, hebben de andere routers de informatie met betrekking tot deze pseudodraad niet nodig. Deze informatie is bijvoorbeeld het label "VC" dat moet worden gebruikt.

Met LDP als het signaalprotocol om pseudodraden in te stellen, wordt de informatie slechts ontvangen door het paar routers, omdat LDP de signaling op een point-to-point manier doet.

Met BGP als het signaalprotocol om pseudodraden in te stellen, wordt de informatie door alle andere routers ontvangen omdat interne BGP (iBGP) de signaling in een point-to-multipoint mode doet. iBGP heeft een volledig vermaasde behoefte, zodat één router een iBGP update naar alle andere iBGP routers stuurt. Dit zou ook kunnen gebeuren met een routereflector.

Als iBGP het signaleringsprotocol is, zullen er twee methoden zijn om updates te verzenden:

1. Elke PE-router adverteert één BGP-update voor alle iBGP-buren voor elke PW; Telkens wordt een MPLS VC-label toegevoegd. Dus zou één PE router evenveel BGP updates

verzenden als er PE routers zijn. Het VC-label dat aan de BGP-update is gekoppeld, kan echter slechts door een van de PE-routers worden gebruikt - de PE-router aan het andere einde van de PW.

2. Om deze kwestie van een hoog aantal BGP updates te vermijden, werd een architectuur ontworpen waarbij één lokale PE router een reeks of een blok van lokale VC etiketten naar alle verre PE routers verstuurt. Elke externe PE-router kiest een van de VC-labels om te gebruiken als een extern VC-label voor de PW naar de lokale PE-router. De externe PE-router moet een extern VC-label op een unieke manier kiezen, zodat geen andere PE-router hetzelfde VC-label kiest uit het geadverteerde blok met labels. Aangezien er een blok etiketten wordt verzonden, moeten er voldoende etiketten beschikbaar zijn om alle mogelijke PW's te kunnen dienen die zouden kunnen worden opgezet, maar er mogen niet zoveel etiketten worden gereserveerd dat ze ongebruikt en verspild zijn.

In dit document wordt beschreven hoe BGP wordt gebruikt om de pseudodraden aan te geven; Let erop dat BGP ook wordt gebruikt voor het tegelijkertijd detecteren van de auto.

Architectuur van de oplossing

Omdat dit VPLS is, is er nog een hop-bij-hop signaleringsprotocol nodig in de kern om de geëtiketteerde pakketten van PE naar PE router te kunnen vervoeren. Deze vervoerfunctie in de kern moet nog steeds worden vervuld door LDP of MPLS traffic engineering.

BGP moet de benodigde informatie doorsturen om de pseudobedrading op een punt-tot-multipoint-manier te kunnen instellen die door VPLS nodig is. Deze signaleringsinformatie omvat:

- PE-routerendpointidentificatie
- VPLS-identiteit
- Blok van MPLS-labels
- Insluitingsinformatie

Identificatie van AE-routerendpoints

De identificatie van het PE-routereindpunt wordt bepaald van de PE-router die de BGP-zender van de update is.

De BGP-update met betrekking tot Layer 2 Virtual Private Networks (L2VPN's) VPLS wordt geïdentificeerd door AFI/SAFI 25/65. Deze adresfamilie wordt overeengekomen wanneer BGP het OPEN-bericht verstuurt.

VPLS-identiteit- en MPLS-labels

Het NLRI, ook bekend als het voorvoegsel, houdt de informatie op de VPLS-identiteit en het blok van MPLS-labels in. Zijn codering heeft een totale lengte van 19 bytes:

```
+-----+
| Length (2 octets) |
+-----+
| Route Distinguisher (8 octets) |
```

```

+-----+
|  VE ID (2 octets)          |
+-----+
|  VE Block Offset (2 octets)|
+-----+
|  VE Block Size (2 octets)  |
+-----+
|  Label Base (3 octets)     |
+-----+

```

De routeswitchmachine (RD) heeft betrekking op de identiteit van de VPLS.

Opmerking: In de Cisco IOS en Cisco NX-OS softwareimplementatie moeten alle PE routers dezelfde RD binnen hetzelfde VPLS-domein hebben.

De virtuele expansie (VE)-ID, VE-blokoffset, VE-blokgrootte en de labelbasis (LB) hebben betrekking op het geadverteerde blok labels, zoals in de volgende sectie wordt uitgelegd.

Insluitingsinformatie

Insluitingsinformatie wordt ook aan het voorvoegsel toegevoegd en wordt gecodeerd als een uitgebreide community 'Layer 2 Info Extended Community' aan de BGP-update. De waarde is 0x800A en wordt gecodeerd als:

```

+-----+
| Extended community type (2 octets) |
+-----+
| Encaps Type (1 octet)              |
+-----+
| Control Flags (1 octet)            |
+-----+
| Layer-2 MTU (2 octet)              |
+-----+
| Reserved (2 octets)                |
+-----+

```

Het type Encaps voor VPLS is 19.

De controlevlakken (bit vector) worden op deze manier gecodeerd:

```

  0 1 2 3 4 5 6 7
+-----+
|  MBZ      |C|S|      (MBZ = MUST Be Zero)
+-----+

```

Name Waarde Betekenis

- | | | |
|---|---|--|
| C | 1 | Er MOET een controle-woord zijn wanneer VPLS-pakketten naar deze PE worden verzonden. |
| | 0 | Er MAG GEEN controle-woord aanwezig zijn wanneer VPLS-pakketten naar deze AE worden verzonden. |
| S | 1 | Er MOET gebruik worden gemaakt van geordende levering van frames wanneer VPLS-pakketten naar deze PE worden verzonden. |
| | 0 | Een geordende levering van frames MAG NIET worden gebruikt wanneer VPLS-pakketten naar deze PE worden verzonden. |

Er zijn ook routedoelstellingen (RT's) verbonden aan de BGP-bijwerking. RT's controleren de

invoer in en de uitvoer uit L2VPN op dezelfde manier als MPLS L3VPN.

VPLS BGP-voorvoegsel voor automatische detectie en VPLS BGP-signaleringsvoorvoegsel

Een voorvoegsel voor automatische ontdekking van VPLS BGP is een voorvoegsel /96, terwijl een voorvoegsel van VPLS BGP-signalering een voorvoegsel /136 is. Dit zijn voorbeelden van elk:

```
PE2#show bgp l2vpn vpls all
BGP table version is 264, local router ID is 10.100.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:100					
*>i 1:100:VEID-1001:Blk-150/136					
	10.100.1.1	0	100	0	?
*> 1:100:10.100.1.2/96					
	0.0.0.0			32768	?

```
PE2#show bgp l2vpn vpls rd 1:100 ve-id 1001 block-offset 150
BGP routing table entry for 1:100:VEID-1001:Blk-150/136, version 262
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
Not advertised to any peer
Refresh Epoch 1
Local
  10.100.1.1 (metric 21) from 10.100.1.4 (10.100.1.4)
  Origin incomplete, metric 0, localpref 100, valid, internal, best
  AGI version(0), VE Block Size(50) Label Base(10105)
  Extended Community: RT:1:100 RT:32:64 L2VPN L2:0x0:MTU-1500
  Originator: 10.100.1.1, Cluster list: 10.100.1.4
  rx pathid: 0, tx pathid: 0x0
```

```
PE2#show bgp l2vpn vpls rd 1:100 10.100.1.2
BGP routing table entry for 1:100:10.100.1.2/96, version 43
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
Not advertised to any peer
Refresh Epoch 1
Local
  0.0.0.0 from 0.0.0.0 (10.100.1.2)
  Origin incomplete, localpref 100, weight 32768, valid, sourced, local,
  best, AGI version(0)
  Extended Community: RT:1:100 L2VPN AGI:1:100
  rx pathid: 0, tx pathid: 0x0
```

Cisco IOS-softwareconfiguratie

Dit is een voorbeeld van Cisco IOS softwareconfiguratie:

```
!
l2vpn vfi context one
vpn id 100
```

```

autodiscovery bgp signaling bgp      <<< "signaling ldp" would be RFC 4762
ve id 1001
ve range 50
route-target export 32:64
route-target import 32:64

mpls label range 10000 20000

!
bridge-domain 1
 member Ethernet0/0 service-instance 100
 member vfi one

!
l2 router-id 10.100.1.1
!

interface Ethernet0/0
 no ip address
 service instance 100 ethernet
!

!
router bgp 1
 bgp log-neighbor-changes
 neighbor 10.100.1.4 remote-as 1
 neighbor 10.100.1.4 update-source Loopback0
!
 address-family l2vpn vpls
 neighbor 10.100.1.4 activate
 neighbor 10.100.1.4 send-community extended
 neighbor 10.100.1.4 suppress-signaling-protocol ldp
 exit-address-family

```

geadvertieerd labelblok

Eén PE-router moet ten minste één labelblok adverteren. Het labelblok is een continue set MPLS-labels en wordt gebruikt door de externe PE-routers om één extern VC-label te selecteren. Het label op afstand wordt gebruikt voor de PW tussen de lokale en externe PE-router. (Een PE-router kan meerdere labelblokken adverteren, zoals uitgelegd in latere secties.)

De VE-ID moet op elke PE worden geconfigureerd. Het identificeert de PE routers binnen het VPLS-domein.

De VE Block Size (VBS) is de grootte van het labelblok en heeft een standaardwaarde van 10. Als 've range' is ingesteld, is het die waarde. 'bereik' kan worden ingesteld als [11-100].

The Label Base (LB) is de eerste labelwaarde van een vrije reeks labels die door de PE-router kunnen worden gereserveerd voor dit VPLS-domein.

VE Block Offset (VBO) is de offset waarde die wordt gebruikt wanneer meerdere labelblokken moeten worden gemaakt door een PE-router. VBO wordt met deze vergelijking berekend: $VBO = \text{RND}(\text{VE-ID}/\text{VBS}) * \text{VBS}$

Dit zijn voorbeelden van berekeningen:

- Indien VBS = 8 en VE-ID = 2, $VBO = \text{RND}(2/8) * 8 = 1$

- Indien VBS = 8 en VE-ID = 20, VBO = RND (20/8) * 8 = 16
- Indien VBS = 50 en VE-ID = 199, VBO = RND (199/50) * 50 = 150
- Indien VBS = 50 en VE-ID = 1002, VBO = RND (1002/50) * 50 = 1000

Het labelblok dat wordt geadverteerd met de externe PE-routers is {LB, LB + 1, ..., LB + VBS - 1}. Het labelblok wordt gedefinieerd door de LB en de VBS; het blok begint bij LB en eindigt bij (LB + VBS - 1).

Er kunnen meerdere labelblokken worden gemaakt door elke PE-router, indien nodig. De router moet ervoor zorgen dat het een ononderbroken reeks vrije etiketten is.

Routeonderscheiding en routedoelstellingen

Voorbeelden van configuratie van PE1

```
router bgp 1

l2vpn vfi context one
  vpn id 100
  autodiscovery bgp signaling bgp
  ve id 1001
  ve range 50
  route-target export 32:64
  route-target import 32:64

mpls label range 10000 20000
```

Dit is een verklaring van de configuratiewaarden:

- De VPN-id is ingesteld op 100.
- De RD is afkomstig van [ASN:vpn-id], tenzij een RD uitdrukkelijk is ingesteld. Hier is de RD 1.100.
- De streefcijfers voor de in- en uitvoerroute zijn 32:64.
- De LB valt binnen de marge [10000 20000]. De exacte waarde van LB hangt af van de eerste reeks gratis continue lokale etiketten die groot genoeg zijn om alle door de VBS bepaalde etiketten te bevatten.
- De VE-ID wordt ingesteld op 1001.
- De VBS is ingesteld op 50.
- De VBO wordt berekend als: $VBO = \text{RND}(\text{VE-ID}/\text{VBS}) * \text{VBS}$ of $\text{RND}(1001/50) * 50 = 1000$.

Label bereik controleren

U kunt het labelbereik controleren met de opdracht **MPLS**-labelbereik:

```
PE1#show mpls label range
Downstream Generic label region: Min/Max label: 10000/20000
```

Er is standaard labelbereik per platform, dat u kunt wijzigen met de opdracht **MPLS**-labelbereik.

Etikels controleren

U kunt de eigenlijke gebruikte labels voor één labelblok in de LFIB-informatiebasis controleren met de opdracht **MPLS-doorvoertabel**.

```
PE1#show mpls forwarding-table
Local      Outgoing  Prefix          Bytes Label  Outgoing  Next Hop  Label
Label      or Tunnel Id   Switched       interface
10000      No Label    lbl-blk-id(1:0) 0           drop
10001      No Label    lbl-blk-id(1:1) 0           drop
10002      No Label    lbl-blk-id(1:2) 0           drop
?
10048      No Label    lbl-blk-id(1:48) 0           drop
10049      No Label    lbl-blk-id(1:49) 0           drop
10050      Pop Label   10.100.1.4/32   0           Et1/0     10.1.1.4
```

In dit voorbeeld, PE1, de lokale router, gereserveerde 50 lokale etiketten voor het etiketblok. "lbl-blk-id(1:0)": een blok-id van 1 en een blokexemplaar van 0, dat het eerste etiket van het blok identificeert. Het laatste etiket van dit blok is label 10049.

De "Uitgaande" interface in de LFIB is "laten vallen" zolang er geen PW is opgezet voor dat lokale label. Als er een PW wordt ingesteld, is de 'Uitgaande' interface 'Geen punt2Point'.

Label Block controleren

Het toegewezen labelblok kan ook met de opdracht van de samenvatting van de **show mpls infrastructuur** blok-database worden gecontroleerd wanneer 'service interne' is geconfigureerd.

```
PE1#show mpls infrastructure lfd block-database summary
Block-DB entry for block-id : 0x1
Block-size : 50, App-Key type : ATOM PWID, Labels : 10000 - 10049
```

LB bedraagt 10000 LB. In dit voorbeeld is het labelblok van LB naar (LB + VBS - 1) of van 10000 naar (10000 + 50 - 1) = 10049.

Geavanceerd prefixeren controleren

U kunt het geadverteerde prefix controleren met de opdracht **bgp l2vpn vpls rd 1:100**:

```
PE1#show bgp l2vpn vpls rd 1:100
BGP table version is 3, local router ID is 10.100.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
      Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 1:100
*>  1:100:VEID-1001:Blk-1000/136
           0.0.0.0                      32768 ?
```

Voorvoegsel in detail bekijken

Om dit voorvoegsel in detail te bekijken, gebruikt u de opdracht **Bgp l2vpn vpls rd 1:100 ve-id 1001 block-offset 1000**. Let op dat u de VE-ID en het label blok specificeert, die in de NLRI (BLK-1000) kunnen worden gevonden.

```
PE1#show bgp l2vpn vpls rd 1:100 ve-id 1001 block-offset 1000
BGP routing table entry for 1:100:VEID-1001:Blk-1000/136, version 3
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
Advertised to update-groups:
  1
Refresh Epoch 1
Local
  0.0.0.0 from 0.0.0.0 (10.100.1.1)
    Origin incomplete, localpref 100, weight 32768, valid, sourced, local, best
    AGI version(0), VE Block Size(50) Label Base(10000)
    Extended Community: RT:1:100 RT:32:64 L2VPN L2:0x0:MTU-1500
    rx pathid: 0, tx pathid: 0x0
```

NLRI geeft de RD van 1:100, de VE-ID van 1001, de VBO van 1000, de VBS van 50, en de LB van 10000.

Layer 2 Info Extended Community heeft deze informatie:

- Encap type is 19 (VPLS)
- Besturingsvlaggen: C = 0 (er mag geen controlewoord worden ingesteld); S = 0 (geen gesequentieerde levering van frames)
- MTU is 1.500

De RT Extended Community heeft deze informatie:

- 1.100 uur
- RT 32:64

Opmerking: Standaard VBS (10) is klein zodat lokale labels niet verspild worden.

Advers, Ontvang en proces van labelblokken in BGP-update-berichten

Wanneer een lokale PE-router een L2VPN-prefix/label blok adverteert, moet elke externe PE-router proberen één label uit dat bereik te kiezen om te kunnen gebruiken als een extern VC-label.

- Als de externe PE-router slaagt, gebruikt deze het label van de externe VC en programmeert deze in het gegevensvlak. BGP heeft geen verdere signalering meer.
- Als de externe PE-router faalt, moet het wachten tot een ander L2VPN VPLS-prefix wordt geadverteerd door de lokale PE-router, dan proberen om een ander extern VC-label uit dat labelblok te kiezen.

Stel dat PE1 een lokale PE is met de vorige configuratie en dat PE2 een afgelegen PE is met deze configuratie:

```
l2vpn vfi context one
vpn id 100
autodiscovery bgp signaling bgp
```

```
ve id 1002
ve range 50
!
mpls label range 3000 60000
```

PE2: Ontvang BGP-update

PE2 ontvangt deze BGP-update van PE1:

```
PE2#show bgp l2vpn vpls rd 1:100 ve-id 1001 block-offset 1000
BGP routing table entry for 1:100:VEID-1001:Blk-1000/136, version 5
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
Not advertised to any peer
Refresh Epoch 2
Local
 10.100.1.1 (metric 21) from 10.100.1.4 (10.100.1.4)
  Origin incomplete, metric 0, localpref 100, valid, internal, best
  AGI version(0), VE Block Size(50) Label Base(10000)
  Extended Community: RT:1:100 RT:32:64 L2VPN L2:0x0:MTU-1500
  Originator: 10.100.1.1, Cluster list: 10.100.1.4
  rx pathid: 0, tx pathid: 0x0
```

PE2: Een label zoeken

PE2 moet een label vinden dat het kan gebruiken als een label van een externe VC voor de PW in de richting van PE1.

PE2 moet eerst bepalen of de VBO binnen het bereik van de configuratie valt. PE2 controleert zijn VE-ID met het door PE1 geadverteerde bereik met de berekening $VBO \leq VE-ID < VBO + VBS$. In dit geval, $1000 \leq 1002 < 1000 + 50$, zodat PE2 slaagt.

PE2 moet dan een label van een externe VC kiezen. Het label demultiplexer (VC) dat door de externe PE moet worden gebruikt, wordt berekend als $(LB + VE-ID - VBO)$.

Vanaf het vorige prefix is LB 10000 en VBO 1000. De VE-ID is die van PE2 en is 1002. Dus het etiket van PE2 plukken $(LB + VE-ID - VBO) = (10000 + 1002 - 1000) = 10002$.

Gebruik de opdracht **show l2vpn vfi één** opdracht om dit te controleren:

```
PE2#show l2vpn vfi name one
Legend: RT=Route-target, S=Split-horizon, Y=Yes, N=No

VFI name: one, state: up, type: multipoint, signaling: BGP
VPN ID: 100, VE-ID: 1002, VE-SIZE: 50
RD: 1:100, RT: 1:100
Bridge-Domain 100 attachment circuits:
Pseudo-port interface: pseudowire100001
Interface      Peer Address    VE-ID  Local Label  Remote Label  S
pseudowire100002  10.100.1.1      1001   3101         10002         Y
```

PE2: Prefixeren bij PE1 verzenden

PE2 stuurt vervolgens het voorvoegsel naar PE1:

```

PE1#show bgp l2vpn vpls rd 1:100 ve-id 1002 block-offset 1000
BGP routing table entry for 1:100:VEID-1002:Blk-1000/136, version 4
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
  Not advertised to any peer
  Refresh Epoch 1
  Local
    10.100.1.2 (metric 21) from 10.100.1.4 (10.100.1.4)
      Origin incomplete, metric 0, localpref 100, valid, internal, best
      AGI version(0), VE Block Size(50) Label Base(3100)
      Extended Community: RT:1:100 L2VPN L2:0x0:MTU-1500
      Originator: 10.100.1.2, Cluster list: 10.100.1.4
      rx pathid: 0, tx pathid: 0x0

```

PE1: Een label zoeken

PE1 is nu de afgelegen PE en moet een label vinden dat het kan gebruiken als een op afstand aangebracht VC-label voor de PW naar PE2.

PE1 moet eerst bepalen of de VBO binnen het bereik van de configuratie valt. PE1 controleert zijn VE-ID met het door PE2 geadverteerde bereik met de berekening $VBO \leq VE-ID < VBO + VBS$. In dit geval slaagt $1000 \leq 1001 < 1000 + 50$, dus PE1.

PE1 moet dan een label van een externe VC kiezen. Het label demultiplexer (VC) dat door de externe PE moet worden gebruikt, wordt berekend als $(LB + VE-ID - VBO)$.

Vanaf het vorige prefix is LB 3100 en VBO 1000. De VE-ID is die van PE1 en is 1001. Dus het etiket van PE1 plukken $(LB + VE-ID - VBO) = (3100 + 1001 - 1000) = 3101$.

Gebruik de opdracht **show l2vpn vfi één** opdracht om dit te controleren:

```

PE1#show l2vpn vfi name one
Legend: RT=Route-target, S=Split-horizon, Y=Yes, N=No

VFI name: one, state: up, type: multipoint, signaling: BGP
VPN ID: 100, VE-ID: 1001, VE-SIZE: 50
RD: 1:100, RT: 1:100, 32:64
Bridge-Domain 1 attachment circuits:
Pseudo-port interface: pseudowire100001

```

Interface	Peer Address	VE-ID	Local Label	Remote Label	S
pseudowire100002	10.100.1.2	1002	10002	3101	Y

Aanvullende verificatieopdrachten

```

PE1#show mpls l2transport vc detail
Local interface: VFI one vfi up
Interworking type is Ethernet
Destination address: 10.100.1.2, VC ID: 100, VC status: up
  Output interface: Et1/0, imposed label stack {17 3101}
  Preferred path: not configured
  Default path: active
  Next hop: 10.1.1.4
Create time: 02:06:08, last status change time: 02:06:08
  Last label FSM state change time: 02:06:08
Signaling protocol: BGP

```

Status TLV support (local/remote) : Not Applicable
LDP route watch : Not Applicable
Label/status state machine : established, LruRru
Last local dataplane status rcvd: No fault
Last BFD dataplane status rcvd: Not Applicable
Last BFD peer monitor status rcvd: Not Applicable
Last local AC circuit status rcvd: No fault
Last local AC circuit status sent: No fault
Last local PW i/f circ status rcvd: No fault
Last local LDP TLV status sent: Not Applicable
Last remote LDP TLV status rcvd: Not Applicable
Last remote LDP ADJ status rcvd: Not Applicable
MPLS VC labels: local 10002, remote 3101
Group ID: local 0, remote 0
MTU: local 1500, remote 1500
Control Word: Off
Dataplane:
SSM segment/switch IDs: 8195/4097 (used), PWID: 3
VC statistics:
transit packet totals: receive 0, send 0
transit byte totals: receive 0, send 0
transit packet drops: receive 0, seq error 0, send 0

PE1#show mpls infrastructure lfd block-database id 1

Block-DB entry for block-id : 0x1

Block-size : 50, App-Key type : ATOM PWID

App-Key entries:

l2ckt(1) 10000
l2ckt(2) 10001
l2ckt(3) 10002
l2ckt(4) 10003
l2ckt(5) 10004
l2ckt(6) 10005
l2ckt(7) 10006
l2ckt(8) 10007
l2ckt(9) 10008
l2ckt(10) 10009
l2ckt(11) 10010
l2ckt(12) 10011
l2ckt(13) 10012
l2ckt(14) 10013
l2ckt(15) 10014
l2ckt(16) 10015
l2ckt(17) 10016
l2ckt(18) 10017
l2ckt(19) 10018
l2ckt(20) 10019
l2ckt(21) 10020
l2ckt(22) 10021
l2ckt(23) 10022
l2ckt(24) 10023
l2ckt(25) 10024
l2ckt(26) 10025
l2ckt(27) 10026
l2ckt(28) 10027
l2ckt(29) 10028
l2ckt(30) 10029
l2ckt(31) 10030
l2ckt(32) 10031
l2ckt(33) 10032
l2ckt(34) 10033
l2ckt(35) 10034
l2ckt(36) 10035

```

12ckt(37) 10036
12ckt(38) 10037
12ckt(39) 10038
12ckt(40) 10039
12ckt(41) 10040
12ckt(42) 10041
12ckt(43) 10042
12ckt(44) 10043
12ckt(45) 10044
12ckt(46) 10045
12ckt(47) 10046
12ckt(48) 10047
12ckt(49) 10048
12ckt(50) 10049

```

```
PE1#show l2vpn atom vc destination 10.100.1.2
```

Interface	Dest Address	VC ID	Service		Status
			Type	Name	
pw100002	10.100.1.2	100	vfi	one	UP

```
PE1#show l2vpn atom vc destination 10.100.1.2 detail
```

```

pseudowire100002 is up, VC status is up PW type: Ethernet
Create time: 02:11:13, last status change time: 02:11:13
Last label FSM state change time: 02:11:13
Destination address: 10.100.1.2 VC ID: 100
Output interface: Et1/0, imposed label stack {17 3101}
Preferred path: not configured
Default path: active
Next hop: 10.1.1.4
Member of vfi service one
Bridge-Domain id: 1
Service id: 0xe7000001
Signaling protocol: BGP
Local VE ID: 1001, Remote VE ID: 1002
Status TLV support (local/remote) : Not Applicable
LDP route watch : Not Applicable
Label/status state machine : established, LruRru
Local dataplane status received : No fault
BFD dataplane status received : Not Applicable
BFD peer monitor status received : Not Applicable
Status received from access circuit : No fault
Status sent to access circuit : No fault
Status received from pseudowire i/f : No fault
Status sent to network peer : Not Applicable
Status received from network peer : Not Applicable
Adjacency status of remote peer : Not Applicable

```

Bindings

Parameter	Local	Remote
Label	10002	3101
Group ID	0	0
Interface		
MTU	1500	1500
Control word	off	off
PW type	Ethernet	Ethernet
VCCV CV type	0x32	0x32
	LSPV [2], BFD/Raw [5]	LSPV [2], BFD/Raw [5]
	BFD/Raw + sig [6]	BFD/Raw + sig [6]
VCCV CC type	0x07	0x07
	CW [1], RA [2], TTL [3]	CW [1], RA [2], TTL [3]
Status TLV	disabled	N/A

```
Dataplane:
  SSM segment/switch IDs: 8195/4097 (used), PWID: 3
Rx Counters
  0 input transit packets, 0 bytes
  0 drops, 0 seq err
Tx Counters
  0 output transit packets, 0 bytes
  0 drops
```

```
PE1#show l2vpn signaling rib rd 1:100
+- Origin of entry (i=iBGP/e=eBGP)
| +- Provisioned (Yes/No)?
| | +- Stale entry (Yes/No)?
| | |
v v v
O P S RD VE-ID VBO VBS LB Next-Hop
-----+-----+-----+-----+-----+-----+
i Y N 1:100 1002 1000 50 3100 10.100.1.2
```

```
PE1#show l2vpn signaling rib rd 1:100 detail
Route 1:100:1002 (epoch:0) from iBGP peer 10.100.1.2
Provisioned (Y) Stale (N)
Route-Target: 1:100
NLRI [FF000001]
  VE-ID:1002 VBO:1000 VBS:50 LB:3100
  MTU: 1500 Control Word: off
RIB Filter [27000002]
  RD: 1:100
  VE-ID: 1001, VBO: 1000, VBS: 50 LB: 10000
Forwarder [58000001] VFI one
```

```
PE1#show l2vpn atom pwid
AToM Pseudowire IDs: In use: 50, In holddown: 0
```

Label	Peer-Address	VCID	PWID	In-Use	FirstUse	ResuedAt	FreedAt
10000	0.0.0.0	0	1	Yes	00:00:15	Never	Never
10001	0.0.0.0	0	2	Yes	00:00:15	Never	Never
10002	10.100.1.2	100	3	Yes	00:00:15	Never	Never
10003	0.0.0.0	0	4	Yes	00:00:15	Never	Never
10004	0.0.0.0	0	5	Yes	00:00:15	Never	Never

```
PE1#show l2vpn atom summary
Destination address: 10.100.1.2, total number of vc: 1
  0 unknown, 1 up, 0 down, 0 admin down, 0 recovering, 0 standby, 0 hotstandby
  1 active vc on MPLS interface Et1/0
```

Meervoudige L2VPN-VPLS prefixes geadverteerd door PE-router voor één VFI

Het is mogelijk dat één PE voor één virtuele expediteur (VFI) meerdere etiketteringsboksen moet bekendmaken.

Als de VE-ID van de afgelegen PE niet valt in het bereik dat door de lokale PE wordt geadverteerd, kan de afgelegen PE geen label voor de PW kiezen. Deze berekening, die eerder is beschreven, is $VBO \leq VE-ID < VBO + VBS$.

Als deze controle faalt, is VE-ID van de afgelegen PE buiten bereik. Het externe PE negeert het voorvoegsel dat van het lokale PE wordt ontvangen. De lokale PE leert dat de afgelegen PE

buiten bereik is wanneer het voorvoegsel wordt ontvangen dat de afgelegen PE reclame is. De lokale PE moet bepalen wat op afstand te gebruiken label voor die op afstand geplaatste PE-router. De lokale PE stuurt ook een nieuw, tweede aanspreekpunt voor een nieuw blok lokale etiketten naar de afgelegen PE, die de afgelegen PE moet kunnen gebruiken om een etiket op afstand te selecteren.

PE1-configuratie

Het vorige voorbeeld wordt hier voortgezet; PE1 heeft nog steeds:

```
l2vpn vfi context one
  vpn id 100
  autodiscovery bgp signaling bgp
  ve id 1001
  ve range 50
  route-target export 32:64
  route-target import 32:64
!
mpls label range 10000 20000
```

PE2-configuratie

PE2 heeft nu een VE-ID van 1002 en deze configuratie:

```
l2vpn vfi context one
  vpn id 100
  autodiscovery bgp signaling bgp
  ve id 10002
  ve range 50
!
mpls label range 3000 60000
```

Oorspronkelijke labelblokken

Zowel PE1 als PE2 beginnen met deze aanvankelijke labelblokken.

```
PE1#show bgp l2vpn vpls rd 1:100
BGP table version is 2, local router ID is 10.100.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
               x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found
```

```
      Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 1:100
 *>  1:100:VEID-1001:Blk-1000/136
           0.0.0.0                               32768 ?
```

```
PE2#show bgp l2vpn vpls rd 1:100
BGP table version is 3, local router ID is 10.100.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
```

x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:100					
*> 1:100:VEID-10002:Blk-10000/136	0.0.0.0			32768	?

PE1 en PE2 uitwisseling

Gebruik de opdracht **debug bgp l2vpn updates** om de PE1 en PE2 uitwisseling te bekijken, dan gebruik de **show bgp l2vpn vpls rd 1:100** opdracht om details te bekijken.

```
PE1#
%BGP-5-ADJCHANGE: neighbor 10.100.1.4 Up
BGP(9): update formatted for 1:100:VEID-1001:Blk-1000:VBS-50:LB-10000/136 VE ID
1001 VE Block Offset 1000 VE Block Size 50 Label Base 10000 /136
BGP(9): (base) 10.100.1.4 send UPDATE (format) 1:100:VEID-1001:Blk-1000:VBS-50:
LB-10000/136, next 10.100.1.1, metric 0, path Local, extended community RT:1:100
RT:32:64 L2VPN L2:0x0:MTU-1500
BGP(9): 10.100.1.4 rcvd UPDATE w/ attr: nexthop 10.100.1.2, origin ?,
localpref 100, metric 0, originator 10.100.1.2, clusterlist 10.100.1.4, extended
community RT:1:100 L2VPN L2:0x0:MTU-1500
BGP(9): 10.100.1.4 rcvd 1:100:VEID-10002:Blk-10000:VBS-50:LB-3000/136
BGP(9): bump net 1:100:VEID-10002:Blk-10000:VBS-50:LB-3000/136, non bpath added
BGP(9): nettable_walker called for 1:100:VEID-10002:Blk-10000:VBS-50:LB-3000/136
BGP(9): best path[0] 1:100:VEID-10002:Blk-10000:VBS-50:LB-3000/136 source
10.100.1.1 nh 10.100.1.2 vpls-id: L2VPN L2:0x0:MTU-1500
BGP(9): add XC RIB route 1:100:VEID-10002:Blk-10000:VBS-50:LB-3000/136 masklen 136
L2VPN L2:0x0:MTU-1500 pathcount: 1 [0] LDP source:10.100.1.1 nexthop:10.100.1.2
RT:1:100
BGP(9): bump net 1:100:VEID-1001:Blk-10000:VBS-50:LB-10053/136, non bpath added
BGP(9): nlri update add VBS 50 LB 10053
BGP(9): nlri update add export extcomm count 4
BGPSSA ssaccount is 0
BGP(9): update formatted for 1:100:VEID-10002:Blk-10000:VBS-50:LB-3000/136 VE ID
10002 VE Block Offset 10000 VE Block Size 50 Label Base 3000 /136
BGP(9): nettable_walker called for 1:100:VEID-1001:Blk-10000:VBS-50:LB-10053/136
BGP(9): nettable_walker 1:100:VEID-1001:Blk-10000:VBS-50:LB-10053/136 route sourced
locally
BGP(9): update formatted for 1:100:VEID-1001:Blk-10000:VBS-50:LB-10053/136 VE ID
1001 VE Block Offset 10000 VE Block Size 50 Label Base 10053 /136
BGP(9): (base) 10.100.1.4 send UPDATE (format) 1:100:VEID-1001:Blk-10000:VBS-50:
LB-10053/136, next 10.100.1.1, metric 0, path Local, extended community RT:1:100
RT:32:64 L2VPN L2:0x0:MTU-1500 L2VPN L2:0x0:MTU-1500
BGP(9): 10.100.1.4 rcvd UPDATE w/ attr: nexthop 10.100.1.2, origin ?, localpref 100,
metric 0, originator 10.100.1.2, clusterlist 10.100.1.4, extended community
RT:1:100 L2VPN L2:0x0:MTU-1500
BGP(9): 10.100.1.4 rcvd 1:100:VEID-10002:Blk-1000:VBS-50:LB-3053/136
BGP(9): bump net 1:100:VEID-10002:Blk-1000:VBS-50:LB-3053/136, non bpath added
BGP(9): nettable_walker called for 1:100:VEID-10002:Blk-1000:VBS-50:LB-3053/136
BGP(9): best path[0] 1:100:VEID-10002:Blk-1000:VBS-50:LB-3053/136 source 10.100.1.1
nh 10.100.1.2 vpls-id: L2VPN L2:0x0:MTU-1500
BGP(9): add XC RIB route 1:100:VEID-10002:Blk-1000:VBS-50:LB-3053/136 masklen 136
L2VPN L2:0x0:MTU-1500 pathcount: 1 [0] LDP source:10.100.1.1 nexthop:10.100.1.2
RT:1:100
BGP(9): update formatted for 1:100:VEID-10002:Blk-1000:VBS-50:LB-3053/136 VE ID
10002 VE Block Offset 1000 VE Block Size 50 Label Base 3053 /136
BGPSSA ssaccount is 0
```


PE1#show bgp l2vpn vpls rd 1:100

BGP table version is 5, local router ID is 10.100.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:100					
*> 1:100:VEID-1001:Blk-1000/136	0.0.0.0			32768	?
*> 1:100:VEID-1001:Blk-10000/136	0.0.0.0			32768	?
*>i 1:100:VEID-10002:Blk-1000/136	10.100.1.2	0	100	0	?
*>i 1:100:VEID-10002:Blk-10000/136	10.100.1.2	0	100	0	?

PE2#show bgp l2vpn vpls rd 1:100

BGP table version is 6, local router ID is 10.100.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 1:100					
*>i 1:100:VEID-1001:Blk-1000/136	10.100.1.1	0	100	0	?
*>i 1:100:VEID-1001:Blk-10000/136	10.100.1.1	0	100	0	?
*> 1:100:VEID-10002:Blk-1000/136	0.0.0.0			32768	?
*> 1:100:VEID-10002:Blk-10000/136	0.0.0.0			32768	?

Analyse van PE1 en PE2-uitwisseling

PE1 en PE2 hebben nu twee merkblokken aan elkaar geadverteerd.

PE1 eerste adverteert met een eerste BGP-update naar PE2:

```
BGP(9): update formatted for 1:100:VEID-1001:Blk-1000:VBS-50:LB-10000/136 VE ID
1001 VE Block Offset 1000 VE Block Size 50 Label Base 10000 /136
BGP(9): (base) 10.100.1.4 send UPDATE (format) 1:100:VEID-1001:Blk-1000:VBS-50:
LB-10000/136, next 10.100.1.1, metric 0, path Local, extended community
RT:1:100 RT:32:64 L2VPN L2:0x0:MTU-1500
```

Bij deze update wordt het NLRI ingesteld volgens de configuratie op PE1.

PE1 ontvangt vervolgens de initiële BGP-update van PE2.

```
BGP(9): 10.100.1.4 rcvd UPDATE w/ attr: nexthop 10.100.1.2, origin ?, localpref
100, metric 0, originator 10.100.1.2, clusterlist 10.100.1.4, extended
community RT:1:100 L2VPN L2:0x0:MTU-1500
BGP(9): 10.100.1.4 rcvd 1:100:VEID-10002:Blk-10000:VBS-50:LB-3000/136
```

PE2 adverteert het voorvoegsel met de waarden VE-ID 10002, VBO = 10000, VBS = 50, LB =

3000.

PE1 merkt op dat PE2 buiten bereik is aangezien PE1 met labelblok LB naar $(LB + VBS - 1)$ of van 1000 naar $(10000 + 50 - 1)$ is begonnen = 10049.

PE1 moet bepalen of de VBO binnen het bereik van de configuratie valt. De VE-ID van PE2 moet dus worden gecontroleerd aan de hand van het door PE1 aangegeven bereik. De berekening is $VBO \leq VE-ID < VBO + VBS$. In dit geval, $1000 \leq 10002 < 1000 + 50$, wat niet waar is. PE1 moet dus een nieuw labelblok sturen om de buiten bereik gelegen VE-ID van PE2 te verwerken. Als reactie op de oorspronkelijke update van PE2, PE1 formaten en stuurt een nieuwe, extra BGP-update naar PE2. PE1 gebruikt nu een nieuwe VBO van 10000.

```
BGP(9): update formatted for 1:100:VEID-1001:Blk-10000:VBS-50:LB-10053/136
VE ID 1001 VE Block Offset 10000 VE Block Size 50 Label Base 10053 /136
BGP(9): (base) 10.100.1.4 send UPDATE (format) 1:100:VEID-1001:Blk-10000:
VBS-50:LB-10053/136, next 10.100.1.1, metric 0, path Local, extended
community RT:1:100 RT:32:64 L2VPN L2:0x0:MTU-1500 L2VPN L2:0x0:MTU-1500
```

Voor PE1 is de VBO 10000, VBS 50, LB 10053. De controle op PE2 is $VBO \leq VE-ID < VBO + VBS$. In dit geval, $10000 \leq 10002 < 10000 + 50$, wat waar is. PE2 kan een etiket op afstand van dit nieuwe label [10053 - 10102] van PE1 kiezen. Met andere woorden, PE1 heeft een nieuw labelblok toegevoegd om PE2 te bereiken en heeft twee BGP-update-berichten gestuurd.

Hetzelfde gebeurt in de tegenovergestelde richting. PE2 ontvangt de oorspronkelijke BGP-bijwerking van PE1. Deze actualisering heeft deze waarden VE-ID 1001, VBO = 1000, VBS = 50, LB = 10000.

PE2 merkt op dat VE-ID van PE1 buiten bereik is met de initiële update van PE2. PE1's check is $VBO \leq VE-ID < VBO + VBS$ of $10000 \leq 1001 < 10000 + 50$. Als reactie verstuurt PE2 deze tweede BGP-update, met een nieuw labelblok [3053 - 3102] dat aan de VE-ID van 110 voldoet 001 van PE1 omdat PE1's check $VBO \leq VE-ID < VBO + VBS$ of $1000 \leq 1001 < 1000 + 50$ is.

```
BGP(9): 10.100.1.4 rcvd UPDATE w/ attr: nexthop 10.100.1.2, origin ?,
localpref 100, metric 0, originator 10.100.1.2, clusterlist 10.100.1.4,
extended community RT:1:100 L2VPN L2:0x0:MTU-1500
BGP(9): 10.100.1.4 rcvd 1:100:VEID-10002:Blk-1000:VBS-50:LB-3053/136
```

Prefixdetails

Dit zijn de details van de twee prefixes van PE1:

```
PE1#show bgp l2vpn vpls rd 1:100 ve-id 1001 block-offset 1000
BGP routing table entry for 1:100:VEID-1001:Blk-1000/136, version 2
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
  Not advertised to any peer
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (10.100.1.1)
      Origin incomplete, localpref 100, weight 32768, valid, sourced, local, best
      AGI version(0), VE Block Size(50) Label Base(10000)
      Extended Community: RT:1:100 RT:32:64 L2VPN L2:0x0:MTU-1500
      rx pathid: 0, tx pathid: 0x0
```

```
PE1#show bgp l2vpn vpls rd 1:100 ve-id 1001 block-offset 10000
```

```
BGP routing table entry for 1:100:VEID-1001:Blk-10000/136, version 4
Paths: (1 available, best #1, table L2VPN-VPLS-BGP-Table)
Not advertised to any peer
Refresh Epoch 1
Local
 0.0.0.0 from 0.0.0.0 (10.100.1.1)
  Origin incomplete, localpref 100, weight 32768, valid, sourced, local, best
  AGI version(0), VE Block Size(50) Label Base(10053)
  Extended Community: RT:1:100 RT:32:64 L2VPN L2:0x0:MTU-1500
    L2VPN L2:0x0:MTU-1500
  rx pathid: 0, tx pathid: 0x0
```

Hier, hebben twee PE routers distiguous number schemes, wat elke PE ertoe brengt om twee BGP updates uit te sturen. Als er veel PE routers met distiguous number schemes zijn, groeit het aantal BGP updates snel zeer groot.

www.cisco.com zegt : "VE-ID-nummersequenties zoals 1, 2, 3 of 501, 502, 503 zijn bijvoorbeeld goed omdat de VE-ID's aaneengesloten zijn. Een nummeringssysteem zoals 100, 200, 300 is slecht omdat het niet aaneengesloten is."

De eerste voorbeelden van 1, 2, 3 of 501, 502, 503 zijn aaneengesloten getallen, dus elke PE-router hoeft slechts één L2VPN VPLS-prefix te verzenden. In het derde voorbeeld (100, 200, 300) moet elke PE veel L2VPN-voorfixes verzenden. Voor niet-contigeeuze getallen zou een groot genoeg VE-bereik het aantal prefixes dat moet worden geadverteerd, lager houden. De hoeveelheid gereserveerde (verspilde) etiketten is echter nog groter.

Interoperabiliteit

Als de BGP Route Reflector (RR) software draait die RFC 4761 niet begrijpt, maar wel ondersteuning heeft voor RFC 4762, is de speciale BGP **buurman x.x.x voorvoegsel-lengte 2** configuratieopdracht nodig in de RFC zodat deze de BGP updates kan weerspiegelen die voor RFC 4761 zijn gebruikt.

Prefixes worden meestal verzonden met een lengte van 1 bytes. Cisco IOS-software implementeerde het concept 'concept-ietf-l2vpn-signalering-08,' dat later RFC 6074 werd. Er is op dat moment een lengte-veld van 1 bytes geselecteerd, dat de lengte in bits aangeeft.

[RFC 6074 Provisioning, automatische detectie en signalering in Layer 2 Virtual Private Networks \(L2VPN's\)](#) specificeert dat de NLRI-codering voor de BGP-automatische detectie een lengte van 2 bytes zou moeten hebben. De 2 bytes geven aan hoeveel bytes van prefix volgen in prefix met variabele lengte.

Sectie 7 van RFC 6074, "BGP-AD en VPLS-BGP Interoperabiliteit", stelt:

"Zowel BGP-AD als VPLS-BGP [RFC4761] gebruiken hetzelfde AFI/SAFI. Om te voorkomen dat zowel BGP-AD als VPLS-BGP naast elkaar bestaan, moet de NLRI-lengte als demultiplexer worden gebruikt.

De BGP-AD NLRI heeft een lengte van NLRI van 12 bytes, die slechts een RD en een VSI-ID van 4 bytes bevat. VPLS-BGP [RFC4761] gebruikt een 17-byte NLRI-lengte. Daarom moeten implementaties van BGP-AD NLRI die groter zijn dan 12 bytes negeren."

Als de opdracht **voor het voorvoegsel x.x.x, groot 2**, niet aanwezig is in de RRs, komt de BGP buurman niet omhoog, en de RR interpreteert het lengteveld slechts als 1 byte. Deze

kennisgeving is te vinden op de RR:

```
%BGP-3-NOTIFICATION: sent to neighbor 10.100.1.2 3/10 (illegal network) 1 bytes FF
%BGP-4-MSGDUMP: unsupported or mal-formatted message received from 10.100.1.2:
FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF 005E 0200 0000 4780 0E1C 0019 4104 0A64
0102 0000 1100 0000 0100 0000 6427 1227 1000 3200 BB80 4001 0102 4002 0080 0404
*Feb 15 12:14:11.561: %BGP_SESSION-5-ADJCHANGE: neighbor 10.100.1.2 L2VPN Vpls
topology base removed from session BGP Notification sent
*Feb 15 12:14:11.561: %BGP_SESSION-5-ADJCHANGE: neighbor 10.100.1.2 IPv4 Unicast
topology base removed from session BGP Notification sent
```

Dit bericht verschijnt op de PE-router:

```
%BGP-3-NOTIFICATION: received from neighbor 10.100.1.4 3/10 (illegal network)
1 bytes FD
```

Dit gebeurt omdat, in de oorspronkelijke implementatie van de automatische ontdekking van BGP in Cisco IOS software, het lengte veld 1 bytes was.

Als u de opdracht voor het voorvoegsel x.x.x.x voorvoegsel-lengte 2 op de RR zet, verschijnen de meldingen niet.

```
router bgp 1
 neighbor 10.100.1.2 remote-as 1
 neighbor 10.100.1.2 update-source Loopback0
!
 address-family l2vpn vpls
 neighbor 10.100.1.2 activate
 neighbor 10.100.1.2 send-community extended
 neighbor 10.100.1.2 prefix-length-size 2
 neighbor 10.100.1.2 route-reflector-client
```