

mVPN에 대한 엄격한 RPF 확인

목차

[소개](#)

[배경 정보](#)

[문제](#)

[솔루션](#)

[Cisco IOS 참고 사항](#)

[구성](#)

[결론](#)

소개

이 문서에서는 mVPN(Multicast over VPN)에 대한 RPF(Strict Reverse Path Forwarding) 기능에 대해 설명합니다. 이 문서에서는 Cisco IOS®의 예제 및 구현¹을 사용하여 동작을 설명합니다.

배경 정보

RPF는 수신 인터페이스가 소스를 향해 검사됨을 의미합니다. 인터페이스가 소스를 향하는 올바른 인터페이스인지 확인하기 위해 확인되지만, 인터페이스가 해당 인터페이스의 올바른 RPF 인접 디바이스인지 확인하기 위해 확인되지 않습니다. 다중 액세스 인터페이스에는 RPF를 수행할 수 있는 인접 디바이스가 두 개 이상 있을 수 있습니다. 그 결과 라우터가 해당 인터페이스에서 동일한 멀티캐스트 스트림의 2배를 수신하고 두 가지를 모두 전달하게 될 수 있습니다.

PIM(Protocol Independent Multicast)이 멀티 액세스 인터페이스에서 실행되는 네트워크에서는 중복 멀티캐스트 스트림으로 인해 어설션 메커니즘이 실행되며 하나의 멀티캐스트 스트림이 더 이상 수신되지 않기 때문에 문제가 되지 않습니다. 경우에 따라 PIM은 멀티 액세스 인터페이스인 MDT(Multicast Distribution Tree)에서 실행되지 않습니다. 이러한 경우 BGP(Border Gateway Protocol)는 오버레이 신호 프로토콜입니다.

PIM이 오버레이 프로토콜로 실행되더라도 분할된 MDT가 있는 프로파일에서 어설션을 수행할 수 없습니다. 이러한 이유는 둘 이상의 Ingress PE 라우터가 있는 시나리오에서 한 PE(Ingress Provider Edge)가 다른 Ingress PE의 Partitioned MDT에 조인하지 않기 때문입니다. 각 인그레스 PE 라우터는 멀티캐스트 트래픽을 볼 수 있는 다른 인그레스 PE 라우터 없이 멀티캐스트 스트림을 분할된 MDT로 전달할 수 있습니다. 두 개의 서로 다른 인그레스 PE 라우터가 동일한 멀티캐스트 스트림에 대해 서로 다른 인그레스 PE 라우터로 각각 MDT를 조인한다는 사실은 유효한 시나리오입니다. Anycast Source라고 합니다. 이렇게 하면 서로 다른 수신기가 동일한 멀티캐스트 스트림에 가입하지만 MPLS(Multiprotocol Label Switching) 코어의 다른 경로를 통해 조인할 수 있습니다. Anycast Source의 예는 그림 1을 참조하십시오.

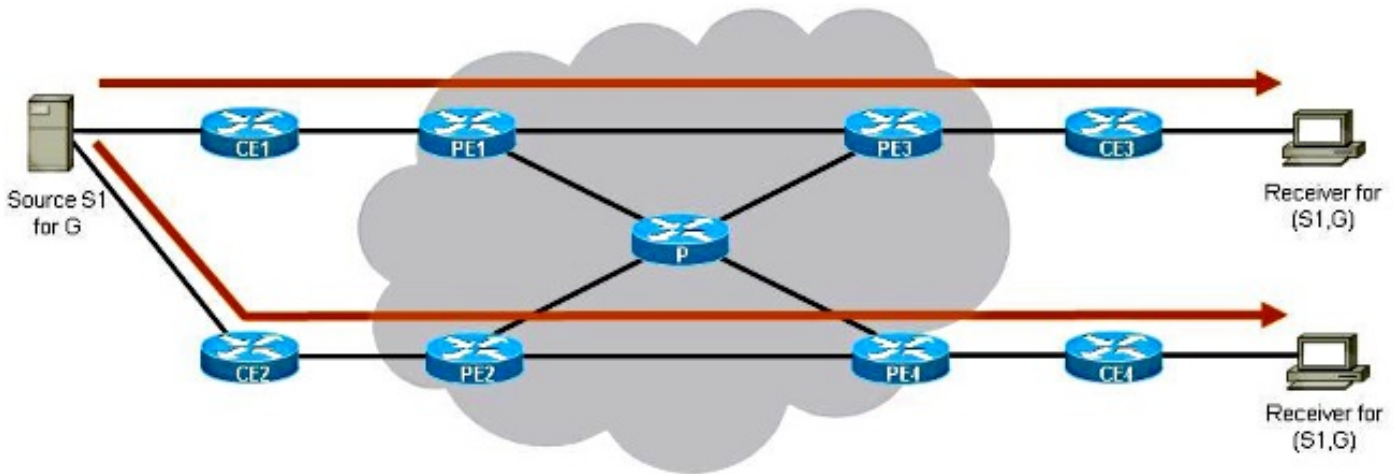


그림 1

Ingress PE 라우터는 두 가지가 있습니다. PE1 및 PE2. 2개의 이그레스 PE 라우터가 있습니다. PE3 및 PE4. 각 이그레스 PE 라우터에는 RPF 인접 디바이스와 다른 인그레스 PE 라우터가 있습니다. PE3에는 RPF 인접 디바이스로 PE1이 있습니다. PE4는 RPF 인접 디바이스로 PE2를 포함합니다. 이그레스 PE 라우터는 가장 가까운 인그레스 PE 라우터를 RPF 인접 라우터로 선택합니다.

스트림(S1,G)은 위쪽 경로를 통해 S1에서 수신기 1로, 아래쪽 경로를 통해 S1에서 수신기 2로 이동합니다. 두 경로를 통해 두 스트림이 교차되지 않습니다(MPLS 코어의 각 경로는 다른 분할된 MDT임).

MDT가 기본 MDT(예: 기본 MDT 프로파일)인 경우 두 멀티캐스트 스트림이 동일한 기본 MDT에 있고 어설션 메커니즘이 실행되므로 이 작업은 작동하지 않습니다. MDT가 기본 MDT 프로파일의 Data MDT인 경우 모든 Ingress PE 라우터는 다른 Ingress PE 라우터의 Data MDT에 조인합니다. 따라서 서로간의 멀티캐스트 트래픽을 볼 수 있으며 어설션 메커니즘이 다시 실행됩니다. 오버레이 프로토콜이 BGP인 경우 UMH(Upstream Multicast Hop)가 선택되고 하나의 Ingress PE 라우터만 전달자로 선택되지만 이는 MDT에 따라 선택됩니다.

Anycast Source는 Partitioned MDT를 실행하는 데 큰 이점 중 하나입니다.

문제

일반 RPF 확인에서는 패킷이 올바른 RPF 인터페이스에서 라우터에 도착하는지 확인합니다. 해당 인터페이스의 올바른 RPF 인접 디바이스에서 패킷이 수신되었는지 확인할 수 없습니다.

그림 2를 참조하십시오. Partitioned MDT가 있는 시나리오에서 중복 트래픽이 지속적으로 전달되는 문제를 표시합니다. Partitioned MDT의 경우 일반 RPF 검사가 중복 트래픽을 방지하기 위해 충분하지 않음을 보여 줍니다.

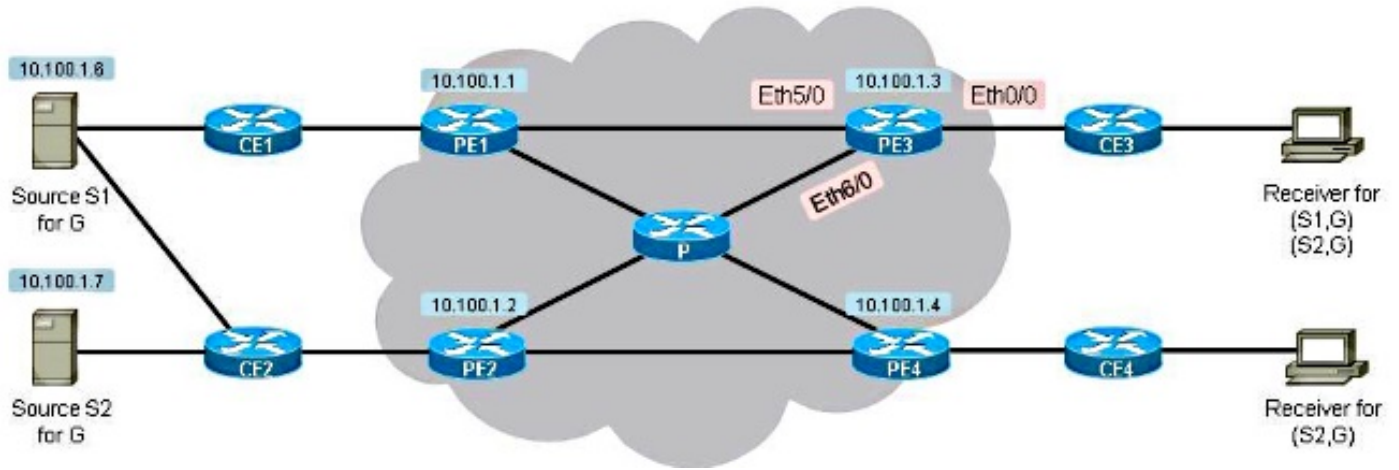


그림 2

수신기가 두 개 있습니다. 첫 번째 수신기가 (S1,G) 및 (S2,G)에 대한 트래픽을 수신하도록 설정됩니다. 두 번째 수신기는 (S2,G)에 대해서만 트래픽을 수신하도록 설정됩니다. 분할된 MDT가 있으며 BGP는 오버레이 신호 프로토콜입니다. 소스 S1은 PE1 및 PE2를 통해 연결할 수 있습니다. 코어 트리 프로토콜은 mLDP(Multipoint Label Distribution Protocol)입니다.

각 PE 라우터는 Type 1 BGP IPv4 mVPN 경로를 광고합니다. 이는 Partitioned MDT의 루트가 될 수 있음을 나타냅니다.

```
PE3#show bgp ipv4 mvpn vrf one
```

```
BGP table version is 257, local router ID is 10.100.1.3
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-pah, 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: 1:3 (default for vrf one)					
*>i [1][1:3][10.100.1.1]/12	10.100.1.1	0	100	0	?
*>i [1][1:3][10.100.1.2]/12	10.100.1.2	0	100	0	?
*> [1][1:3][10.100.1.3]/12	0.0.0.0			32768	?
*>i [1][1:3][10.100.1.4]/12	10.100.1.4	0	100	0	?

PE3는 S1에 대한 유니캐스트 경로를 조회한 후 PE1을 S1에 대한 RPF 인접 디바이스로 찾습니다.

```
PE3#show bgp vpnv4 unicast vrf one 10.100.1.6/32
```

```
BGP routing table entry for 1:3:10.100.1.6/32, version 16
Paths: (2 available, best #2, table one)
Advertised to update-groups:
 5
Refresh Epoch 2
65001, imported path from 1:2:10.100.1.6/32 (global)
 10.100.1.2 (metric 21) (via default) from 10.100.1.5 (10.100.1.5)
   Origin incomplete, metric 0, localpref 100, valid, internal
   Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.2:1
   Originator: 10.100.1.2, Cluster list: 10.100.1.5
   mpls labels in/out nolabel/20
   rx pathid: 0, tx pathid: 0
```

```
Refresh Epoch 2
65001, imported path from 1:1:10.100.1.6/32 (global)
10.100.1.1 (metric 11) (via default) from 10.100.1.5 (10.100.1.5)
  Origin incomplete, metric 0, localpref 100, valid, internal, best
  Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.1:1
  Originator: 10.100.1.1, Cluster list: 10.100.1.5
  mpls labels in/out nolabel/29
  rx pathid: 0, tx pathid: 0x0
```

```
PE3#show ip rpf vrf one 10.100.1.6
```

```
RPF information for ? (10.100.1.6)
```

```
RPF interface: Lspvif0
```

```
RPF neighbor: ? (10.100.1.1)
```

```
RPF route/mask: 10.100.1.6/32
```

```
RPF type: unicast (bgp 1)
```

```
Doing distance-preferred lookups across tables
```

```
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE3는 PE1을 (S1,G)에 대한 RPF 인접 디바이스로 선택하고 PE1을 루트로 분할된 MDT를 조인합니다. PE3는 PE2를 (S2,G)에 대한 RPF 인접 디바이스로 선택하고 PE2를 루트로 분할된 MDT를 조인합니다.

```
PE3#show bgp vpnv4 unicast vrf one 10.100.1.7/32
```

```
BGP routing table entry for 1:3:10.100.1.7/32, version 18
```

```
Paths: (1 available, best #1, table one)
```

```
Advertised to update-groups:
```

```
6
```

```
Refresh Epoch 2
```

```
65002, imported path from 1:2:10.100.1.7/32 (global)
```

```
10.100.1.2 (metric 21) (via default) from 10.100.1.5 (10.100.1.5)
```

```
Origin incomplete, metric 0, localpref 100, valid, internal, best
```

```
Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.2:1
```

```
Originator: 10.100.1.2, Cluster list: 10.100.1.5
```

```
mpls labels in/out nolabel/29
```

```
rx pathid: 0, tx pathid: 0x0
```

```
PE3#show ip rpf vrf one 10.100.1.7
```

```
RPF information for ? (10.100.1.7)
```

```
RPF interface: Lspvif0
```

```
RPF neighbor: ? (10.100.1.2)
```

```
RPF route/mask: 10.100.1.7/32
```

```
RPF type: unicast (bgp 1)
```

```
Doing distance-preferred lookups across tables
```

```
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE4는 (S1,G)에 대한 RPF 인접 디바이스로 PE2를 선택하고 PE1을 루트로 분할된 MDT를 조인합니다.

```
PE4#show bgp vpnv4 unicast vrf one 10.100.1.6/32
```

```
BGP routing table entry for 1:4:10.100.1.6/32, version 138
```

```
Paths: (2 available, best #1, table one)
```

```
Advertised to update-groups:
```

```
2
```

```
Refresh Epoch 2
```

```
65001, imported path from 1:2:10.100.1.6/32 (global)
```

```
10.100.1.2 (metric 11) (via default) from 10.100.1.5 (10.100.1.5)
```

```
Origin incomplete, metric 0, localpref 100, valid, internal, best
```

```
Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.2:1
```

```
Originator: 10.100.1.2, Cluster list: 10.100.1.5
```

```
mpls labels in/out nolabel/20
```

```
rx pathid: 0, tx pathid: 0x0
```

Refresh Epoch 2

```
65001, imported path from 1:1:10.100.1.6/32 (global)
 10.100.1.1 (metric 21) (via default) from 10.100.1.5 (10.100.1.5)
  Origin incomplete, metric 0, localpref 100, valid, internal
  Extended Community: RT:1:1 MVPN AS:1:0.0.0.0 MVPN VRF:10.100.1.1:1
  Originator: 10.100.1.1, Cluster list: 10.100.1.5
  mpls labels in/out nolabel/29
  rx pathid: 0, tx pathid: 0
```

PE4#show ip rpf vrf one 10.100.1.6

RPF information for ? (10.100.1.6)

RPF interface: Lspvif0

RPF neighbor: ? (10.100.1.2)

RPF route/mask: 10.100.1.6/32

RPF type: unicast (bgp 1)

Doing distance-preferred lookups across tables

RPF topology: ipv4 multicast base, originated from ipv4 unicast base

RPF 인터페이스는 S1(10.100.1.6) 및 S2(10.100.1.7)에 모두 Lspvif0입니다.

PE3은 (S2,G)에 대해 PE2에서 분할된 MDT를 조인하고, PE4는 (S1,G)에 대해 PE2에서 분할된 MDT에 조인합니다. PE1은 (S1,G)에 대해 PE1에서 분할된 MDT에 조인합니다. PE1 및 PE2에서 수신된 유형 7 BGP IPv4 mVPN 경로로 이를 확인할 수 있습니다.

PE1#show bgp ipv4 mvpn vrf one

BGP table version is 302, 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: 1:1 (default for vrf one)					
*>i [7][1:1][1][10.100.1.6/32][232.1.1.1/32]/22	10.100.1.3	0	100	0	?

PE2#show bgp ipv4 mvpn vrf one

BGP table version is 329, 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: 1:2 (default for vrf one)					
*>i [7][1:2][1][10.100.1.6/32][232.1.1.1/32]/22	10.100.1.4	0	100	0	?
*>i [7][1:2][1][10.100.1.7/32][232.1.1.1/32]/22	10.100.1.3	0	100	0	?

PE3 및 PE4의 멀티캐스트 항목:

PE3#show ip mroute vrf one 232.1.1.1

Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,

Y - Joined MDT-data group, y - Sending to MDT-data group,
G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
Q - Received BGP S-A Route, q - Sent BGP S-A Route,
V - RD & Vector, v - Vector, p - PIM Joins on route,
x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.7, 232.1.1.1), 21:18:24/00:02:46, flags: sTg
Incoming interface: Lspvif0, **RPF nbr 10.100.1.2**
Outgoing interface list:
Ethernet0/0, Forward/Sparse, 00:11:48/00:02:46

(10.100.1.6, 232.1.1.1), 21:18:27/00:03:17, flags: sTg
Incoming interface: Lspvif0, **RPF nbr 10.100.1.1**
Outgoing interface list:
Ethernet0/0, Forward/Sparse, 00:11:48/00:03:17

PE4#show ip mroute vrf one 232.1.1.1

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
L - Local, P - Pruned, R - RP-bit set, F - Register flag,
T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
U - URD, I - Received Source Specific Host Report,
Z - Multicast Tunnel, z - MDT-data group sender,
Y - Joined MDT-data group, y - Sending to MDT-data group,
G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
Q - Received BGP S-A Route, q - Sent BGP S-A Route,
V - RD & Vector, v - Vector, p - PIM Joins on route,
x - VxLAN group
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(10.100.1.6, 232.1.1.1), 20:50:13/00:02:37, flags: sTg
Incoming interface: Lspvif0, **RPF nbr 10.100.1.2**
Outgoing interface list:
Ethernet0/0, Forward/Sparse, 20:50:13/00:02:37

PE3는 PE1에 루팅된 P2MP(Point-to-Multipoint) 트리와 PE2에 루팅된 트리에도 조인합니다.

PE3#show mpls mldp database

* Indicates MLDP recursive forwarding is enabled

LSM ID : A Type: P2MP Uptime : 00:18:40
FEC Root : 10.100.1.1
Opaque decoded : [gid 65536 (0x00010000)]
Opaque length : 4 bytes
Opaque value : 01 0004 00010000
Upstream client(s) :
10.100.1.1:0 [Active]
Expires : Never Path Set ID : A
Out Label (U) : None Interface : Ethernet5/0*
Local Label (D): 29 Next Hop : 10.1.5.1
Replication client(s):
MDT (VRF one)
Uptime : 00:18:40 Path Set ID : None
Interface : Lspvif0

```
LSM ID : B   Type: P2MP   Uptime : 00:18:40
FEC Root      : 10.100.1.2
Opaque decoded   : [gid 65536 (0x00010000)]
Opaque length    : 4 bytes
Opaque value     : 01 0004 00010000
Upstream client(s) :
  10.100.1.5:0 [Active]
    Expires      : Never          Path Set ID : B
    Out Label (U) : None          Interface   : Ethernet6/0*
    Local Label (D) : 30          Next Hop    : 10.1.3.5
Replication client(s):
  MDT (VRF one)
    Uptime       : 00:18:40      Path Set ID : None
    Interface    : Lspvif0
```

이것은 PE4가 PE2에 루팅된 P2MP 트리에 조인하는 것을 보여줍니다.

PE4#**show mpls mldp database**

* Indicates MLDP recursive forwarding is enabled

```
LSM ID : 3   Type: P2MP   Uptime : 21:17:06
FEC Root      : 10.100.1.2
Opaque decoded   : [gid 65536 (0x00010000)]
Opaque value     : 01 0004 00010000
Upstream client(s) :
  10.100.1.2:0 [Active]
    Expires      : Never          Path Set ID : 3
    Out Label (U) : None          Interface   : Ethernet5/0*
    Local Label (D) : 29          Next Hop    : 10.1.6.2
Replication client(s):
  MDT (VRF one)
    Uptime       : 21:17:06      Path Set ID : None
    Interface    : Lspvif0
```

10pps가 있는 그룹 232.1.1.1의 S1 및 S2 스트림. PE3 및 PE4에서 스트림을 볼 수 있습니다. 그러나 PE3에서는 (S1,G)의 속도를 20pps로 볼 수 있습니다.

PE3#**show ip mroute vrf one 232.1.1.1 count**

Use "show ip mfib count" to get better response time for a large number of mroutes.

IP Multicast Statistics

3 routes using 1692 bytes of memory

2 groups, 1.00 average sources per group

Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second

Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.1.1.1, Source count: 2, Packets forwarded: 1399687, Packets received: 2071455

Source: 10.100.1.7/32, Forwarding: 691517/10/28/2, Other: 691517/0/0

Source: 10.100.1.6/32, Forwarding: 708170/20/28/4, Other: 1379938/671768/0

PE4#**show ip mroute vrf one 232.1.1.1 count**

Use "show ip mfib count" to get better response time for a large number of mroutes.

IP Multicast Statistics

2 routes using 1246 bytes of memory

2 groups, 0.50 average sources per group

Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second

Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.1.1.1, Source count: 1, Packets forwarded: 688820, Packets received:

688820

Source: 10.100.1.6/32, Forwarding: 688820/10/28/2, Other: 688820/0/0

```
PE3#show interfaces ethernet0/0 | include rate
Queueing strategy: fifo
30 second input rate 0 bits/sec, 0 packets/sec
30 second output rate 9000 bits/sec, 30 packets/sec
```

중복된 스트림이 있습니다.이 중복은 PE1에서 분할된 MDT 및 PE2에서 분할된 MDT에 스트림 (S1,G)이 존재한 결과입니다. 이 두 번째 분할된 MDT는 PE2에서 PE3에 조인되어 스트림(S2,G)을 가져옵니다. 그러나 PE4가 (S1,G)를 얻기 위해 PE2에서 분할된 MDT에 조인했기 때문에 (S1,G)도 PE2에서 분할된 MDT에 존재합니다. 따라서 PE3는 연결된 분할된 MDT에서 스트림(S1,G)을 수신합니다.

PE3는 PE1과 PE2에서 수신하는 (S1,G)에 대한 패킷을 구별할 수 없습니다. 두 스트림은 올바른 RPF 인터페이스에서 수신됩니다.Lspvif0

```
PE3#show ip multicast vrf one mpls vif
```

Interface	Next-hop	Application	Ref-Count	Table / VRF name	Flags
Lspvif0	0.0.0.0	MDT	N/A	1 (vrf one) 0x1	

패킷은 PE3의 서로 다른 수신 물리적 인터페이스 또는 동일한 인터페이스에 도착할 수 있습니다.어떤 경우든 (S1,G)에 대한 다른 스트림의 패킷은 PE3에 다른 MPLS 레이블과 함께 도착합니다.

```
PE3#show mpls forwarding-table vrf one
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Label	Outgoing interface	Next Hop
29	[T] No Label	[gid 65536 (0x00010000)][V]	768684		aggregate/one	\
30	[T] No Label	[gid 65536 (0x00010000)][V]	1535940		aggregate/one	\

[T] Forwarding through a LSP tunnel.
View additional labelling info with the 'detail' option

솔루션

그 해결책은 더 엄격한 RPF를 갖는 것이다.엄격한 RPF를 사용하면 라우터는 RPF 인터페이스에서 패킷이 수신되는 인접 디바이스를 확인합니다.엄격한 RPF가 없으면 수신 인터페이스가 RPF 인터페이스인지 확인하는 것뿐이지만, 해당 인터페이스의 올바른 RPF 인접 디바이스에서 패킷을 수신하는지 여부는 확인하지 않습니다.

Cisco IOS 참고 사항

다음은 Cisco IOS를 사용한 RPF에 대한 몇 가지 중요한 참고 사항입니다.

- strict RPF 모드로/에서 변경할 경우 Partitioned MDT를 구성하기 전에 이를 구성하거나 BGP를 지웁니다.strict RPF 명령만 구성하면 다른 Lspvif 인터페이스가 즉시 생성되지 않습니다.
- Cisco IOS에서는 Strict RPF가 기본적으로 활성화되지 않습니다.
- 기본 MDT 프로파일과 함께 strict-rpf 명령을 사용할 수 없습니다.

구성

VRF(Virtual Routing and Forwarding)에 대해 PE3에서 엄격한 RPF를 구성할 수 있습니다.

```
vrf definition one
rd 1:3
!
address-family ipv4
mdt auto-discovery mldp
  mdt strict-rpf interface
  mdt partitioned mldp p2mp
mdt overlay use-bgp
route-target export 1:1
route-target import 1:1
exit-address-family
!
```

RPF 정보가 변경되었습니다.

```
PE3#show ip rpf vrf one 10.100.1.6
RPF information for ? (10.100.1.6)
  RPF interface: Lspvif0
Strict-RPF interface: Lspvif1
  RPF neighbor: ? (10.100.1.1)
RPF route/mask: 10.100.1.6/32
RPF type: unicast (bgp 1)
Doing distance-preferred lookups across tables
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

```
PE3#show ip rpf vrf one 10.100.1.7
RPF information for ? (10.100.1.7)
  RPF interface: Lspvif0
Strict-RPF interface: Lspvif2
  RPF neighbor: ? (10.100.1.2)
RPF route/mask: 10.100.1.7/32
RPF type: unicast (bgp 1)
Doing distance-preferred lookups across tables
RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

PE3는 인그레스 PE당 Lspvif 인터페이스를 생성했습니다. Lspvif 인터페이스는 인그레스 PE, AF(Address Family) 및 VRF별로 생성됩니다. 10.100.1.6의 RPF가 Lspvif1 인터페이스를 가리키고 10.100.1.7의 RPF가 Lspvif2 인터페이스를 가리킵니다.

```
PE3#show ip multicast vrf one mpls vif
```

Interface	Next-hop	Application	Ref-Count	Table / VRF name	Flags
Lspvif0	0.0.0.0	MDT	N/A	1 (vrf one)	0x1
Lspvif1	10.100.1.1	MDT	N/A	1 (vrf one)	0x1
Lspvif2	10.100.1.2	MDT	N/A	1 (vrf one)	0x1

이제 PE1의 패킷(S1,G)에 대한 RPF 검사는 RPF 인터페이스 Lspvif1에 대해 확인됩니다. 이러한 패킷은 MPLS 레이블 29와 함께 제공됩니다. PE2의 패킷(S2,G)에 대한 RPF 검사는 RPF 인터페이스 Lspvif2에 대해 확인됩니다. 이러한 패킷은 MPLS 레이블 30과 함께 제공됩니다. 스트림은 PE3에 도착하지만 다른 인터페이스를 통해 도착할 수 있습니다. 동일한 인터페이스를 제공합니다. 그러나 mLDP가 Penultimate-Hop-Popting(PHP)을 사용하지 않기 때문에 멀티캐스트 패킷 위에 항상 일반 MPLS 레이블이 있습니다. PE1에서 PE2에서 도착한 (S1,G) 패킷은 서로 다른 두 개의 분할된 MDT에 있으므로 다른 MPLS 레이블을 가집니다. 따라서 PE3는 PE1에서 오는 (S1,G) 스트림과 PE2에서 오는 (S1,G) 스트림을 구별할 수 있습니다. 이렇게 하면 PE3에서 패킷을 분리할 수 있으

며 다른 Ingress PE 라우터에서 RPF를 수행할 수 있습니다.

이제 PE3의 mLDP 데이터베이스에 Ingress PE당 서로 다른 Lspvif 인터페이스가 표시됩니다.

```
PE3#show mpls mldp database
```

```
* Indicates MLDP recursive forwarding is enabled
```

```
LSM ID : C   Type: P2MP   Uptime : 00:05:58
```

```
FEC Root      : 10.100.1.1
```

```
Opaque decoded : [gid 65536 (0x00010000)]
```

```
Opaque length  : 4 bytes
```

```
Opaque value   : 01 0004 00010000
```

```
Upstream client(s) :
```

```
10.100.1.1:0 [Active]
```

```
Expires       : Never           Path Set ID : C
```

```
Out Label (U) : None           Interface  : Ethernet5/0*
```

```
Local Label (D): 29           Next Hop   : 10.1.5.1
```

```
Replication client(s):
```

```
MDT (VRF one)
```

```
Uptime       : 00:05:58       Path Set ID : None
```

```
Interface    : Lspvif1
```

```
LSM ID : D   Type: P2MP   Uptime : 00:05:58
```

```
FEC Root      : 10.100.1.2
```

```
Opaque decoded : [gid 65536 (0x00010000)]
```

```
Opaque length  : 4 bytes
```

```
Opaque value   : 01 0004 00010000
```

```
Upstream client(s) :
```

```
10.100.1.5:0 [Active]
```

```
Expires       : Never           Path Set ID : D
```

```
Out Label (U) : None           Interface  : Ethernet6/0*
```

```
Local Label (D): 30           Next Hop   : 10.1.3.5
```

```
Replication client(s):
```

```
MDT (VRF one)
```

```
Uptime       : 00:05:58       Path Set ID : None
```

```
Interface    : Lspvif2
```

인그레스 PE당 엄격한 RPF 또는 RPF는 멀티캐스트 스트림이 인그레스 PE당 다른 MPLS 레이블이 있는 인그레스 PE로 들어오기 때문에 작동합니다.

```
PE3#show mpls forwarding-table vrf one
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Label	Outgoing interface	Next Hop
29	[T] No Label	[gid 65536 (0x00010000)][V]	\	162708	aggregate/one	
30	[T] No Label	[gid 65536 (0x00010000)][V]	\	162750	aggregate/one	

```
[T] Forwarding through a LSP tunnel.
```

```
View additional labelling info with the 'detail' option
```

엄격한 RPF가 작동한다는 증거는 PE3에 더 이상 중복 스트림(S1,G)이 전달되지 않는다는 것입니다. 중복 스트림은 PE3에 도착하지만 RPF 오류로 인해 삭제됩니다. RPF 오류 카운터는 676255이며 10pps의 비율로 계속 증가합니다.

```
PE3#show ip mroute vrf one 232.1.1.1 count
```

```
Use "show ip mfib count" to get better response time for a large number of mroutes.
```

```
IP Multicast Statistics
```

```
3 routes using 1692 bytes of memory
```

2 groups, 1.00 average sources per group

Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second

Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)

Group: 232.1.1.1, Source count: 2, Packets forwarded: 1443260, Packets received: 2119515

Source: 10.100.1.7/32, Forwarding: 707523/10/28/2, Other: 707523/0/0

Source: 10.100.1.6/32, Forwarding: 735737/10/28/2, Other: 1411992/676255/0

이제 PE3의 출력 속도는 20pps로 각 스트림(S1,G) 및 (S2,G)에 대해 10pps입니다.

```
PE3#show interfaces ethernet0/0 | include rate
```

```
Queueing strategy: fifo
```

```
30 second input rate 0 bits/sec, 0 packets/sec
```

```
30 second output rate 6000 bits/sec, 20 packets/sec
```

결론

분할된 MDT를 사용하는 mVPN 구축 모델에는 엄격한 RPF 검사를 사용해야 합니다.

분할된 MDT를 사용하여 mVPN 구축 모델에 대해 엄격한 RPF 검사를 구성하지 않더라도 작업이 작동하는 것처럼 보일 수 있습니다. 멀티캐스트 스트림은 수신자에게 전달됩니다. 그러나 소스가 여러 Ingress PE 라우터에 연결될 때 중복 멀티캐스트 트래픽이 발생할 가능성이 있습니다. 이로 인해 네트워크의 대역폭이 낭비되고 수신기의 멀티캐스트 애플리케이션에 부정적인 영향을 미칠 수 있습니다. 따라서 분할된 MDT를 사용하는 mVPN 구축 모델에 대해 엄격한 RPF 검사를 구성해야 합니다.