

PIM 어설션 메커니즘 이해

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[PIM 어설션 메커니즘이란 무엇입니까?](#)

[시나리오 1. LHR 근거](#)

[RFC 7761 섹션 4.2.2의 요약입니다.](#)

[시나리오 2. 경로 선택 어설션](#)

[RFC 7761 섹션 4.6.3의 요약입니다.](#)

[요약](#)

소개

이 문서에서는 PIM(Protocol Independent Multicast) 어설션 메커니즘을 설명하고 PIM 어설션 승차 기준을 중점적으로 다루며 특정 코너 케이스에 더 깊이 파고들어 봅니다.

사전 요구 사항

요구 사항

Cisco에서는 PIM 어설션 메커니즘에 대해 알고 있는 것이 좋습니다.

사용되는 구성 요소

이 문서의 정보는 Cisco CSR1000V 버전 16.4.1을 기반으로 합니다.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 이해해야 합니다.

PIM 어설션 메커니즘이란 무엇입니까?

공유 세그먼트에 여러 PIM 지원 라우터가 있는 경우 이러한 라우터에 중복 멀티캐스트 트래픽이 발생할 수 있습니다. 동일한 공유 세그먼트에 있는 두 개 이상의 라우터에 동일한 소스 IP/대상 그룹에 대한 공유 세그먼트 쪽으로 나가는 인터페이스를 채우는 유효한 (S,G) 또는 (*,G) 항목이 있을 수 있으므로 이러한 경우가 발생할 수 있습니다.

PIM 어설션 메커니즘은 공유 세그먼트에서 멀티캐스트 트래픽의 중복을 탐지하고 제거하는 데 사용됩니다. 이 메커니즘은 중복이 발생하는 것을 방지하지 않으며 대신 멀티캐스트 트래픽의 중복을 트리거로 사용하여 이 스트림에 대해 단일 전달자를 선택하는 이 메커니즘을 활성화한다는 점에 유의해야 합니다.

공유 세그먼트에서 멀티캐스트 트래픽이 중복되는 경우 공유 세그먼트에 동일한(S,G) 또는 (*,G)을 전송하는 여러 라우터가 있다고 가정할 수 있습니다. 이 스트림을 효과적으로 전달하려면 하나의 라우터를 선택하면 중복이 제거됩니다.

PIM은 OIL(Outgoing Interface List)에서 멀티캐스트 패킷을 수신할 때 트리거되는 PIM 어설션 메시지를 활용합니다. 이러한 어설션 메시지에는 누가 승자가 될 것인지 계산하는 데 사용되는 메트릭이 포함되어 있습니다. LAN의 다운스트림 라우터에서도 PIM 어설션 메시지를 수신합니다. 그런 다음 이러한 메시지는 다운스트림 디바이스에서 어설션 선택을 받은 업스트림 라우터로 적절한 가입/정리 메시지를 보내는 데 사용됩니다.

시나리오 1. LHR 근거

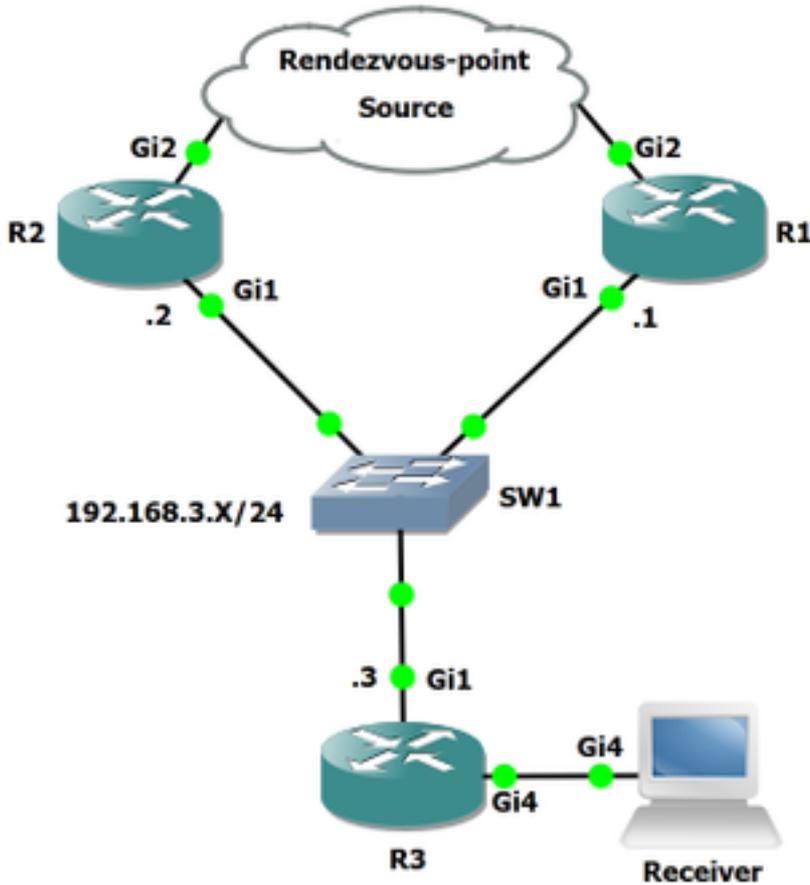


그림 1.

네트워크 다이어그램에서 R3은 LHR(Last Hop Router)이며, R3은 공유 세그먼트를 통해 R2와 R1에 모두 연결됩니다.

수신기에서 IGMP(Internet Group Management Protocol) 보고서를 수신하면 R3은 RPF 인접 디바이스가 RP를 향하는 대상인지 확인합니다. 토폴로지에서 R1은 RPF 인접 디바이스이므로 R3은 R1에 (*,G) 조인을 보냅니다. R1이 스트림을 끌어오면(그룹이 활성 상태인 것으로 가정) R3은 소스 쪽으로 (S,G) 조인을 전송하고 소스 트리를 아래로 가져옵니다. R2는 소스 트리를 향하는 RPF 인접 디바이스입니다. 즉 R3은 R2에 (S,G) 조인을 보냅니다. R3은 RP와 소스 모두에 대해 동일한 RPF 인터페이스를 가집니다. 여기서 그룹 239.1.1.1에 대한 R3 mroute 테이블을 볼 수 있습니다.

```
R3#show ip mroute
IP Multicast Routing Table
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
```

```
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 239.1.1.1), 00:00:55/stopped, RP 192.168.0.100, flags: SJC
  Incoming interface: GigabitEthernet1, RPF nbr 192.168.3.1
  Outgoing interface list:
    GigabitEthernet4, Forward/Sparse, 00:00:55/00:02:04
(10.0.0.2, 239.1.1.1), 00:00:52/00:02:07, flags: JT
  Incoming interface: GigabitEthernet1, RPF nbr 192.168.3.2, Mroute
  Outgoing interface list:
    GigabitEthernet4, Forward/Sparse, 00:00:52/00:02:07
(*, 224.0.1.40), 00:01:22/00:02:09, RP 192.168.0.100, flags: SJPCL
  Incoming interface: GigabitEthernet1, RPF nbr 192.168.3.1
```

R3에서 볼 수 있듯이 (*,G) RPF 인접 디바이스가 192.168.3.1이고 (S,G) 쪽에 있는 RPF 인접 디바이스가 192.168.3.2입니다. 그러면 R1 및 R2가 모두 R3에 적합한 OIL을 갖게 됩니다. 다음 항목을 살펴보겠습니다.

R1#show ip mroute

```
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 239.1.1.1), 00:15:02/00:02:33, RP 192.168.0.100, flags: S
  Incoming interface: GigabitEthernet2, RPF nbr 192.168.5.2
  Outgoing interface list:
    GigabitEthernet1, Forward/Sparse, 00:15:02/00:02:33
(10.0.0.2, 239.1.1.1), 00:13:24/00:02:33, flags: PR
  Incoming interface: GigabitEthernet2, RPF nbr 192.168.5.2
  Outgoing interface list: Null
(*, 224.0.1.40), 00:29:17/00:02:51, RP 192.168.0.100, flags: SJCL
  Incoming interface: GigabitEthernet2, RPF nbr 192.168.5.2
  Outgoing interface list:
    GigabitEthernet1, Forward/Sparse, 00:16:06/00:02:51
  Outgoing interface list: Null
```

R2#show ip mroute

```
IP Multicast Routing Table
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 239.1.1.1), 00:08:00/stopped, RP 192.168.0.100, flags: SP
  Incoming interface: GigabitEthernet2, RPF nbr 192.168.4.1
  Outgoing interface list: Null
(10.0.0.2, 239.1.1.1), 00:00:03/00:02:56, flags: T
  Incoming interface: GigabitEthernet2, RPF nbr 192.168.4.1
  Outgoing interface list:
    GigabitEthernet1, Forward/Sparse, 00:00:03/00:03:26
(*, 224.0.1.40), 01:37:30/00:02:22, RP 192.168.0.100, flags: SJPL
  Incoming interface: GigabitEthernet2, RPF nbr 192.168.4.1
```

앞서 언급했듯이 공유 세그먼트에 유효한 OIL이 채워진 업스트림 라우터가 두 개 있는 경우 어설션

을 트리거할 수 있습니다. R1과 R2 모두 유효한 OIL이 있으므로 패킷 캡처에 어설션 메커니즘이 있는지 확인합니다.

이 패킷 캡처는 R3 인터페이스 Gi1에서 SW1로 캡처되었습니다.

The screenshot shows a Wireshark capture of traffic on the SW1 Ethernet2 to R3 Gi1 interface. The packet list pane displays 25 captured packets. The first 11 packets are OSPF Hello packets from various sources (192.168.3.1, 192.168.3.2, 192.168.3.3) to destination 224.0.0.5. Packets 12 through 25 are ICMP Echo (ping) requests from source 10.0.0.2 to destination 239.1.1.1. The packet details pane at the bottom shows the raw packet data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.3.1	224.0.0.5	OSPF	98	Hello Packet
2	0.705389	192.168.3.2	224.0.0.5	OSPF	98	Hello Packet
3	3.124776	192.168.3.3	224.0.0.5	OSPF	98	Hello Packet
4	7.733948	192.168.3.2	224.0.0.13	PIMv2	72	Hello
5	9.480827	192.168.3.1	224.0.0.5	OSPF	98	Hello Packet
6	10.256987	192.168.3.2	224.0.0.5	OSPF	98	Hello Packet
7	11.954130	192.168.3.1	224.0.0.13	PIMv2	72	Hello
8	12.621371	192.168.3.3	224.0.0.13	PIMv2	72	Hello
9	13.015136	192.168.3.3	224.0.0.5	OSPF	98	Hello Packet
10	19.046520	192.168.3.1	224.0.0.5	OSPF	98	Hello Packet
11	19.670571	192.168.3.2	224.0.0.5	OSPF	98	Hello Packet
12	22.114741	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=0/0, ttl=253 (multicast)
13	22.137371	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
14	22.137597	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
15	22.972394	192.168.3.3	224.0.0.5	OSPF	98	Hello Packet
16	23.085520	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=1/256, ttl=253 (multicast)
17	24.087827	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=2/512, ttl=253 (multicast)
18	24.723777	192.168.3.3	224.0.0.13	PIMv2	96	Join/Prune
19	25.088340	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=3/768, ttl=253 (multicast)
20	26.091246	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=4/1024, ttl=253 (multicast)
21	27.091219	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=5/1280, ttl=253 (multicast)
22	28.109058	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=6/1536, ttl=253 (multicast)
23	29.000065	192.168.3.1	224.0.0.5	OSPF	98	Hello Packet
24	29.118436	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=7/1792, ttl=253 (multicast)
25	29.225379	192.168.3.2	224.0.0.5	OSPF	98	Hello Packet

이 패킷 캡처에서는 R1, R2 및 R3 사이의 공유 세그먼트에 복제를 생성하기 위한 모든 전제 조건이 있지만 어설션 패킷이 표시되지 않습니다. (S,G) 스트림이 활성화되었을 때 PIM 어설션 패킷이 표시되지 않는 이유는 무엇입니까?

RFC 7761이 이러한 질문에 대한 답을 제시할 수 있습니다.

RFC 7761 섹션 4.2.2의 요약입니다.

4.2.2. Setting and Clearing the (S,G) SPTbit

Basically, Update_SPTbit(S,G,iif) will set the SPTbit if we have the appropriate (S,G) join state, and if the packet arrived on the correct upstream interface for S, and if one or more of the following conditions apply:

1. The source is directly connected, in which case the switch to the SPT is a no-op.
2. The RPF interface to S is different from the RPF interface to the RP. The packet arrived on RPF_interface(S), and so the SPT must have been completed.
3. No one wants the packet on the RP tree.
4. $RPF'(S,G) == RPF'(*,G)$. In this case, the router will never be able to tell if the SPT has been completed, so it should just switch immediately. The $RPF'(S,G) != NULL$ check ensures that the SPTbit is set only if the RPF neighbor towards S is valid.

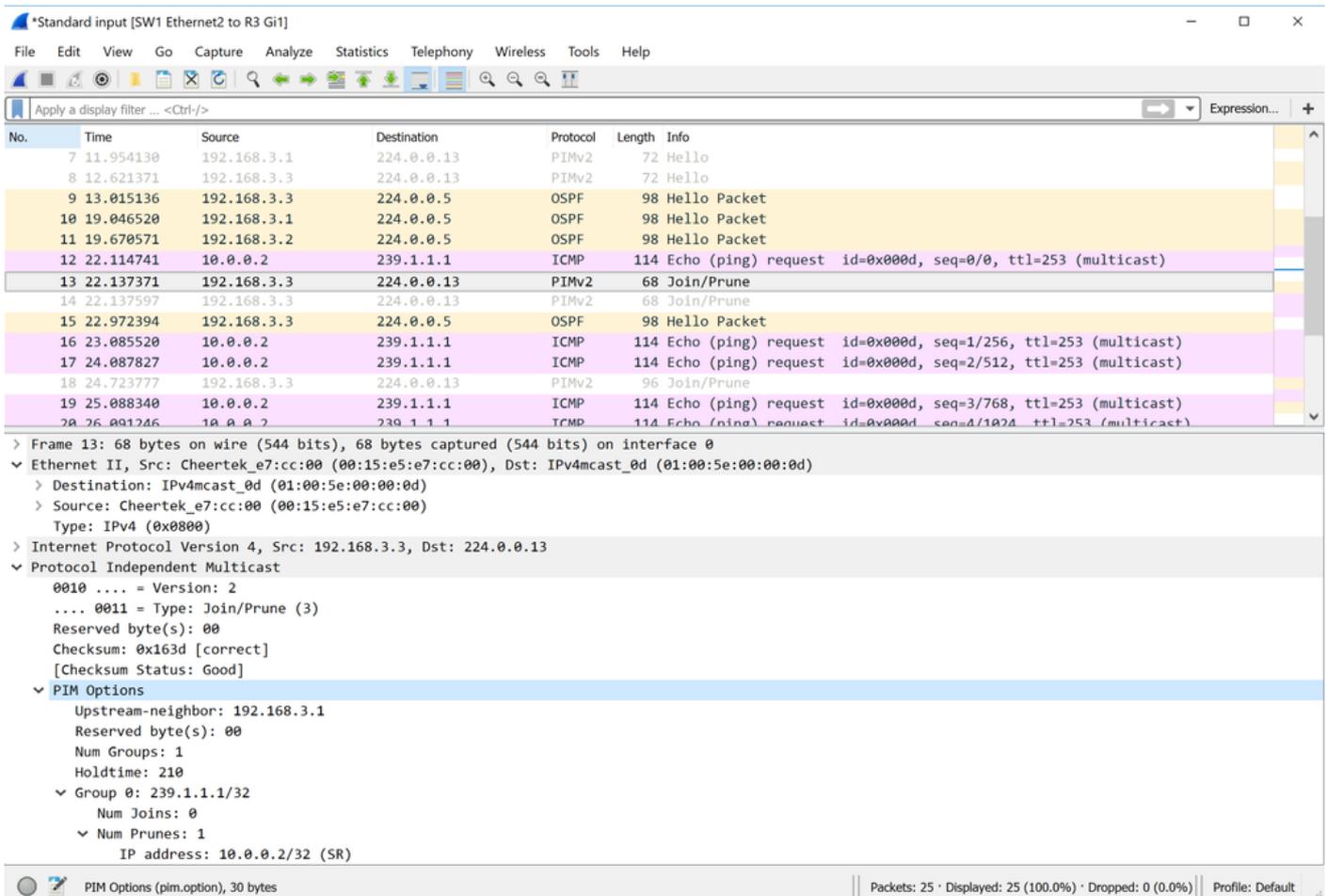
In the case where the RPF interface is the same for the RP and for S, but RPF'(S,G) and RPF'(*,G) differ, we wait for an Assert(S,G), which indicates that the upstream router with (S,G) state believes the SPT has been completed.

(S,G) SPTbit는 (*,G) 또는 on (S,G) 상태를 구분하는 데 사용됩니다. RP 트리에서 소스 트리로 전환할 경우 업스트림(*,G) 상태가 설정되고 업스트림(S,G) 상태가 설정되면 라우터는 (*,G) 상태에서만 계속 전달되어야 합니다. 이렇게 하면 업스트림(S,G) 상태가 설정되기 전에 prune(S,G,rpt)을 전송함으로써 발생하는 임시 블랙홀이 방지됩니다.

시나리오가 위에서 언급한 마지막 포인트와 상관관계를 가질 수 있는 것처럼 보이지만, RPF 인터페이스가 RP와 S에 대해 동일한 경우 그러나 RPF(S,G) 및 RPF'(*,G)가 다르므로 Assert(S,G)(S,G) 상태의 업스트림 라우터가 SPT가 완료되었다고 믿는다는 것을 나타내는 Assert(S,G)를 기다립니다.

어설션을 트리거하려면 라우터가 세그먼트의 동일한 소스 IP/대상 그룹에 대해 이미 채워진 OIL에서 중복 패킷을 수신해야 합니다. R3은 LHR이기도 합니다. 즉, (*,G)에서 패킷을 수신할 때 (*,G)에서 SPT(S,G)로 전환하도록 지정됩니다.

패킷 캡처에서 어떤 어설션도 트리거되지 않음을 확인합니다. 첫 번째 ICMP 에코를 수신한 후 바로 전송된 prune이 표시되지만



No.	Time	Source	Destination	Protocol	Length	Info
7	11.954130	192.168.3.1	224.0.0.13	PIMv2	72	Hello
8	12.621371	192.168.3.3	224.0.0.13	PIMv2	72	Hello
9	13.015136	192.168.3.3	224.0.0.5	OSPF	98	Hello Packet
10	19.046520	192.168.3.1	224.0.0.5	OSPF	98	Hello Packet
11	19.670571	192.168.3.2	224.0.0.5	OSPF	98	Hello Packet
12	22.114741	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=0/0, ttl=253 (multicast)
13	22.137371	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
14	22.137597	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
15	22.972394	192.168.3.3	224.0.0.5	OSPF	98	Hello Packet
16	23.085520	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=1/256, ttl=253 (multicast)
17	24.087827	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=2/512, ttl=253 (multicast)
18	24.723777	192.168.3.3	224.0.0.13	PIMv2	96	Join/Prune
19	25.088340	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=3/768, ttl=253 (multicast)
20	26.001246	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000d, seq=4/1024, ttl=253 (multicast)

```

> Frame 13: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface 0
  > Ethernet II, Src: Cheertek_e7:cc:00 (00:15:e5:e7:cc:00), Dst: IPv4mcast_0d (01:00:5e:00:00:0d)
    > Destination: IPv4mcast_0d (01:00:5e:00:00:0d)
    > Source: Cheertek_e7:cc:00 (00:15:e5:e7:cc:00)
    Type: IPv4 (0x0800)
  > Internet Protocol Version 4, Src: 192.168.3.3, Dst: 224.0.0.13
  > Protocol Independent Multicast
    0010 .... = Version: 2
    ... 0011 = Type: Join/Prune (3)
    Reserved byte(s): 00
    Checksum: 0x163d [correct]
    [Checksum Status: Good]
  > PIM Options
    Upstream-neighbor: 192.168.3.1
    Reserved byte(s): 00
    Num Groups: 1
    Holdtime: 210
    > Group 0: 239.1.1.1/32
      Num Joins: 0
      > Num Prunes: 1
        IP address: 10.0.0.2/32 (SR)
  
```

PIM Options (pim.option), 30 bytes

Packets: 25 · Displayed: 25 (100.0%) · Dropped: 0 (0.0%) Profile: Default

보시다시피 R3 인터페이스 G1에서 첫 번째 ICMP(Internet Control Message Protocol) 요청 패킷을 수신하면 업스트림 인접 디바이스 192.168.3.1으로 (*,G) SR 비트 prune이 전송됩니다. 이 prune(*,G)은 정의된 특정 소스에 대해 생성됩니다.

다음과 같은 플래그도 설정할 수 있습니다. (SR):

The S flag: indicates that the multicast group is a sparse mode group.

The R flag: The R flag is the RP-bit flag and indicates that the information in the (S, G) entry is applicable to the shared tree.

두 번째 PIM 패킷 No. 14에서 R3이 (S,G) 트리에 참가하려고 시도하는 것을 확인할 수 있습니다.

The screenshot shows a Wireshark capture of network traffic on interface 0. The packet list pane highlights packet 14, which is a PIMv2 Join/Prune message. The packet details pane is expanded to show the following structure:

- Frame 14: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface 0
- Ethernet II, Src: Cheertek_e7:cc:00 (00:15:e5:e7:cc:00), Dst: IPv4mcast_0d (01:00:5e:00:00:0d)
 - Destination: IPv4mcast_0d (01:00:5e:00:00:0d)
 - Source: Cheertek_e7:cc:00 (00:15:e5:e7:cc:00)
 - Type: IPv4 (0x0800)
- Internet Protocol Version 4, Src: 192.168.3.3, Dst: 224.0.0.13
- Protocol Independent Multicast
 - 0010 = Version: 2
 - ... 0011 = Type: Join/Prune (3)
 - Reserved byte(s): 00
 - Checksum: 0x173c [correct]
 - [Checksum Status: Good]
 - PIM Options
 - Upstream-neighbor: 192.168.3.2
 - Reserved byte(s): 00
 - Num Groups: 1
 - Holdtime: 210
 - Group 0: 239.1.1.1/32
 - Num Joins: 1
 - IP address: 10.0.0.2/32 (S)
 - Num Prunes: 0

첫 번째 데이터 플레인이 수신되면 패킷 R3이 (*,G)를 제거하고 (S,G)를 구축하는 것으로 관찰됩니다. 이것이 PIM 어설션 패킷이 표시되지 않는 이유입니다. 이 특정 시나리오는 (S,G) 및 (*,G)에 대해 동일한 RPF 인터페이스가 있는 LHR이 있는 경우에 적용됩니다. 이 동작은 RFC 7761과 약간 다를 수 있지만 어떤 문제도 발생시키지 않아야 합니다.

시나리오 2를 계속 진행하겠습니다. 이 시나리오의 다이어그램은 다음과 같습니다.

시나리오 2. 경로 선택 어설션

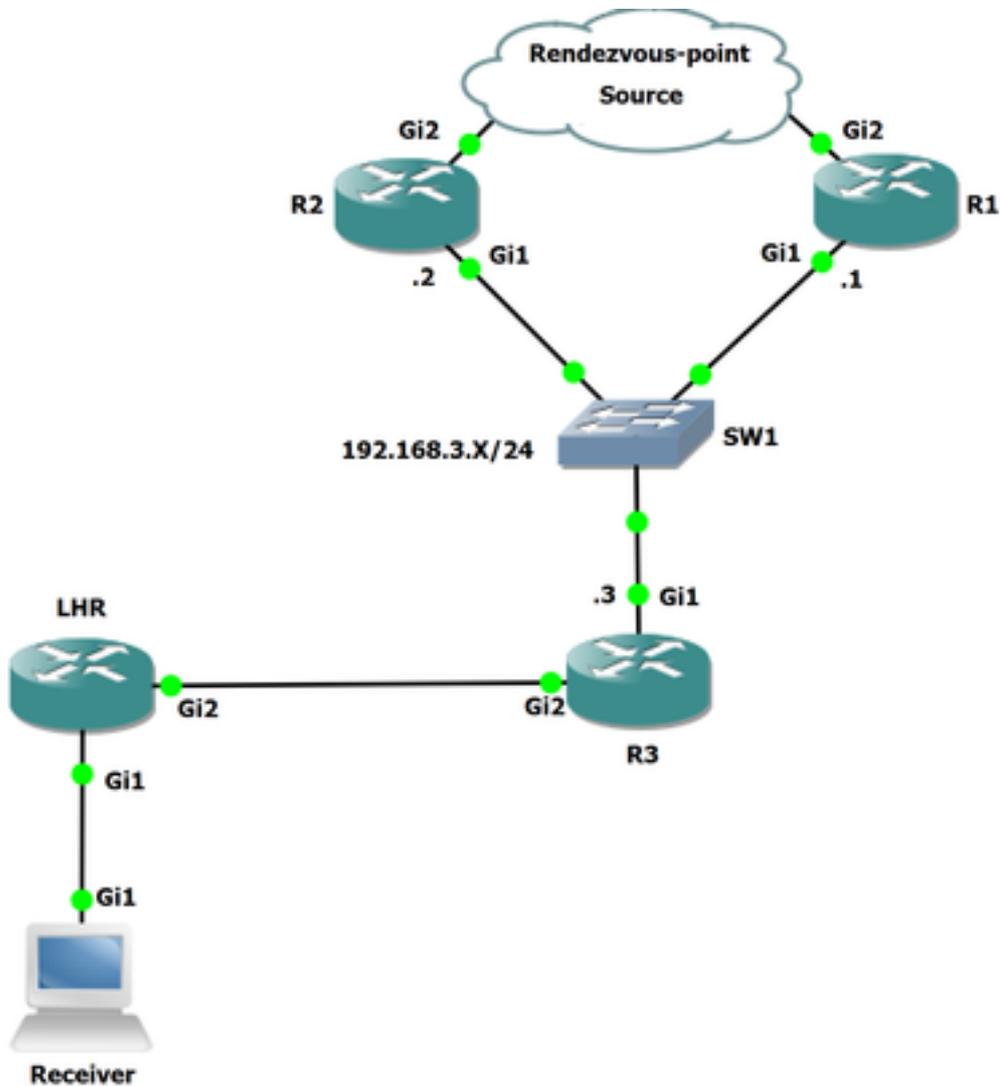


그림 2.

이 토폴로지에서는 LHR인 R3에 연결된 다른 라우터가 있습니다. LHR은 수신자에 직접 연결됩니다. 소스 및 RP는 R2와 R1 둘 다 위에 있습니다. R3에서 RP를 향하는 RPF 네이버는 R1이고 소스를 향하는 RPF 네이버는 R2입니다.

소스 및 RP 모두에 대한 RPF 인접 디바이스를 확인합니다.

다음은 RP를 향하는 RPF 네이버입니다. 192.168.0.100은 192.168.3.1입니다.

```
R3#show ip rpf 192.168.0.100
RPF information for ? (192.168.0.100)
  RPF interface: GigabitEthernet1
  RPF neighbor: ? (192.168.3.1)
  RPF route/mask: 192.168.0.100/32
  RPF type: unicast (ospf 1)
  Doing distance-preferred lookups across tables
  RPF topology: ipv4 multicast base, originated from ipv4 unicast base
```

다음은 Source(소스)에 대한 RPF 네이버입니다. 10.0.0.2은 192.168.3.2입니다.

```
R3#show ip rpf 10.0.0.2
RPF information for ? (10.0.0.2)
  RPF interface: GigabitEthernet1
```

```

RPF neighbor: ? (192.168.3.2)
RPF route/mask: 10.0.0.0/24
RPF type: unicast (ospf 1)
Doing distance-preferred lookups across tables
RPF topology: ipv4 multicast base, originated from ipv4 unicast base

```

소스를 활성화하기 전에 R3의 mroute 테이블을 살펴보겠습니다. 그룹 239.1.1.1에 대해 이미 (*,G)이 있습니다. 이는 LHR에 연결된 수신자가 지정된 그룹에 대해 이미 요청했기 때문입니다.

```

R3#show ip mroute
IP Multicast Routing Table
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.1.1.1), 00:00:57/00:02:32, RP 192.168.0.100, flags: S
  Incoming interface: GigabitEthernet1, RPF nbr 192.168.3.1
  Outgoing interface list:
    GigabitEthernet2, Forward/Sparse, 00:00:57/00:02:32

(*, 224.0.1.40), 00:11:24/00:02:41, RP 192.168.0.100, flags: SJCL
  Incoming interface: GigabitEthernet1, RPF nbr 192.168.3.1
  Outgoing interface list:
    GigabitEthernet2, Forward/Sparse, 00:02:02/00:02:41

```

이제 R3 인터페이스 Gi1에서 소스를 활성화하고 패킷을 캡처합니다.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.3.1	224.0.0.5	OSPF	98	Hello Packet
2	3.164783	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
3	5.264729	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
4	7.447012	192.168.3.2	224.0.0.5	OSPF	98	Hello Packet
5	8.150289	192.168.3.3	224.0.0.5	OSPF	98	Hello Packet
6	9.674810	192.168.3.1	224.0.0.5	OSPF	98	Hello Packet
7	12.016714	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000f, seq=0/0, ttl=253 (multicast)
8	12.166782	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
9	13.974441	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000f, seq=1/256, ttl=253 (multicast)
10	13.975383	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000f, seq=1/256, ttl=253 (multicast)
11	13.980084	192.168.3.1	224.0.0.13	PIMv2	62	Assert
12	13.980901	192.168.3.2	224.0.0.13	PIMv2	62	Assert
13	15.976508	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000f, seq=2/512, ttl=253 (multicast)
14	16.865001	192.168.3.3	224.0.0.13	PIMv2	68	Join/Prune
15	17.334577	192.168.3.2	224.0.0.5	OSPF	98	Hello Packet
16	17.987218	10.0.0.2	239.1.1.1	ICMP	114	Echo (ping) request id=0x000f, seq=3/768, ttl=253 (multicast)
17	18.032846	192.168.3.3	224.0.0.5	OSPF	98	Hello Packet

```

> Frame 11: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0
> Ethernet II, Src: Cheertek_9c:3a:00 (00:15:e5:9c:3a:00), Dst: IPv4mcast_0d (01:00:5e:00:00:0d)
> Internet Protocol Version 4, Src: 192.168.3.1, Dst: 224.0.0.13
  Protocol Independent Multicast
    0010 .... = Version: 2
    ... 0101 = Type: Assert (5)
    Reserved byte(s): 00
    Checksum: 0x5e6a [correct]
    [Checksum Status: Good]
  PIM Options
    Group: 239.1.1.1/32
    Source: 10.0.0.2
    1... .... = RP Tree: True
    .000 0000 0000 0000 0000 0000 0110 1110 = Metric Preference: 110
    Metric: 2

```

이 패킷 캡처에서 볼 수 있듯이 PIM 어설션 패킷은 이미 존재합니다.

프레임 11:

```
> Frame 11: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0
> Ethernet II, Src: Cheertek_9c:3a:00 (00:15:e5:9c:3a:00), Dst: IPv4mcast_0d (01:00:5e:00:00:0d)
> Internet Protocol Version 4, Src: 192.168.3.1, Dst: 224.0.0.13
v Protocol Independent Multicast
  0010 .... = Version: 2
  .... 0101 = Type: Assert (5)
  Reserved byte(s): 00
  Checksum: 0x5e6a [correct]
  [Checksum Status: Good]
  v PIM Options
    Group: 239.1.1.1/32
    Source: 10.0.0.2
    1... .... = RP Tree: True
    .000 0000 0000 0000 0000 0000 0110 1110 = Metric Preference: 110
    Metric: 2
```

프레임 12:

```
> Frame 12: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0
> Ethernet II, Src: Cheertek_8b:3e:00 (00:15:e5:8b:3e:00), Dst: IPv4mcast_0d (01:00:5e:00:00:0d)
> Internet Protocol Version 4, Src: 192.168.3.2, Dst: 224.0.0.13
v Protocol Independent Multicast
  0010 .... = Version: 2
  .... 0101 = Type: Assert (5)
  Reserved byte(s): 00
  Checksum: 0xde6a [correct]
  [Checksum Status: Good]
  v PIM Options
    Group: 239.1.1.1/32
    Source: 10.0.0.2
    0... .... = RP Tree: False
    .000 0000 0000 0000 0000 0000 0110 1110 = Metric Preference: 110
    Metric: 2
```

이러한 패킷을 볼 때 어설션 승자가 누구인지 확인할 수 있어야 합니다. 이제 PIM 어설션 전달자 선택을 살펴보겠습니다.

메트릭 기본 설정은 AD(관리 거리)입니다. 이는 라우팅 테이블에 경로를 설치하는 라우팅 프로토콜의 관리 거리를 말하며, 소스 IP 주소를 조회하는 데 사용되며 메트릭은 경로의 비용입니다.

어설션 승자를 결정하는 데 사용되는 다른 속성도 있습니다. RFC 7761에서 이러한 세부 정보를 볼 수 있습니다.

RFC 7761 섹션 4.6.3의 요약입니다.

4.6.3. Assert Metrics

Assert metrics are defined as:

```
struct assert_metric {
    rpt_bit_flag;
    metric_preference;
    route_metric;
    ip_address;
};
```

When comparing assert_metrics, the rpt_bit_flag, metric_preference,

and route_metric fields are compared in order, where the first lower value wins. If all fields are equal, the primary IP address of the router that sourced the Assert message is used as a tie-breaker, with the highest IP address winning.

이러한 필드를 정의하고 경로를 선택하면 이 시나리오에서 어설션 승자가 누구인지 결정할 수 있습니다. 어설션 패킷을 다시 보면 rpt_bit_flag의 첫 번째 선택 기준에 따라 결정되므로 메트릭 환경 설정이 비교되지 않는 것을 확인할 수 있습니다.

이 시나리오에서는 R1과 R2를 비교합니다. 두 라우터 모두 이전에 확인되었던 어설션 메시지를 전송하고 두 디바이스 모두 서로의 어설션 메시지를 확인하면 서로 메트릭을 비교하여 승자를 확인할 수 있습니다.

R2는 RP 트리와 함께 어설션 메시지를 전송하므로 값이 0인 False는 RP 트리와 함께 보낸 R1보다 훨씬 낮습니다. 값이 1인 True입니다. RP 트리 비트는 0 또는 1로 설정됩니다.

1로 설정된 RP 트리 비트는 현재 공유 트리에 있음을 의미합니다. RPT 비트가 지워진 경우 assert의 발신자가 인터페이스에 (S,G) 전달 상태를 가지고 있음을 나타냅니다.

(S,G) 어설션은 (*,G) 어설션보다 우선순위가 높으므로 R2는 어설션 승자가 되어야 합니다. "I am Assert Winner" 상태로 전환됩니다. RFC 7761의 이전 문에서 언급했듯이, 더 낮은 값이 더 선호됩니다.

R1과 R2를 모두 살펴 누가 주장의 승자를 가려낼 수 있는지 알아보겠습니다.

R2#show ip mroute

IP Multicast Routing Table

Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join

Timers: Uptime/Expires

Interface state: Interface, Next-Hop or VCD, State/Mode

(* , 239.1.1.1), 00:42:52/stopped, RP 192.168.0.100, flags: SP

Incoming interface: GigabitEthernet2, RPF nbr 192.168.4.1

Outgoing interface list: Null

(10.0.0.2, 239.1.1.1), 00:42:52/00:01:40, flags: T

Incoming interface: GigabitEthernet2, RPF nbr 192.168.4.1

Outgoing interface list:

GigabitEthernet1, Forward/Sparse, 00:42:52/00:03:07, A

(* , 224.0.1.40), 00:43:23/00:02:25, RP 192.168.0.100, flags: SJPL

Incoming interface: GigabitEthernet2, RPF nbr 192.168.4.1

Outgoing interface list: Null

이 출력에서 R2의 S,G(S,G)에 OIL에 A 플래그가 설정되어 있으며 이 플래그는 어설션 승자임을 나타냅니다. R1에서는 (S,G)에 OIL이 없고 이 경우 특정(S,G)가 삭제되었음을 의미하는 P 플래그가 설정됩니다. 어설션 승자가 아닙니다.

참고: 공유 세그먼트에 assert가 있는 경우 다운스트림 인접 디바이스는 Join(*,G) 및 Join(S,G) 정기 메시지를 해당 RPF 인접 디바이스(즉, 어설션 프로세스에서 수정한 RPF 인접 디바이스)로 보냅니다. MRIB에 표시된 대로 항상 RPF 인접 디바이스로 전송되지는 않습니다

R1#show ip mroute

IP Multicast Routing Table

Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join

Timers: Uptime/Expires

Interface state: Interface, Next-Hop or VCD, State/Mode

(* , 239.1.1.1), 00:44:32/00:03:09, RP 192.168.0.100, flags: S

Incoming interface: GigabitEthernet2, RPF nbr 192.168.5.2

Outgoing interface list:

GigabitEthernet1, Forward/Sparse, 00:44:32/00:03:09, A

(10.0.0.2, 239.1.1.1), 00:44:19/00:03:09, flags: PR

Incoming interface: GigabitEthernet2, RPF nbr 192.168.5.2

Outgoing interface list: Null

(* , 224.0.1.40), 00:44:50/00:02:53, RP 192.168.0.100, flags: SJCL

Incoming interface: GigabitEthernet2, RPF nbr 192.168.5.2

Outgoing interface list:

GigabitEthernet1, Forward/Sparse, 00:43:56/00:02:53

R1과 R2 모두 RP 트리 비트가 1로 설정된 경우 AD가 가장 낮은 라우터를 고려할 수 있습니다. 동일할 경우 메트릭을 확인합니다. 두 라우터에서 RP 트리 비트가 true이면 메트릭을 RP IP 주소로 비교합니다. RP 트리 비트가 0이면 메트릭을 멀티캐스트 스트림의 소스에 비교합니다.

이러한 값이 모두 동일하면 가장 높은 IP 주소 소싱 어설션 메시지가 낙찰자입니다.

요약

시나리오 1에서는 어설션 패킷을 관찰하지 않았지만 RFC에 따라 트리거되어야 합니다. 앞서 언급했듯이 R3이 (S,G)에 대한 컨트롤 플레인 구축하기 전에 잘라내기(*,G)를 했기 때문입니다.

시나리오 2에서는 어설션 패킷을 볼 수 있습니다. LHR에서 첫 번째 패킷을 수신하면 소스/그룹을 가져오기 위해 R3로 (S,G) join/prune을 보냅니다. 그런 다음 R3는 동일한 소스/그룹에 대해 R2로 조인/prune 패킷을 보냅니다. 그러면 R1과 R2 모두 유효한 OIL을 채울 수 있습니다. 이제 R3s(S,G) 상태에 T 플래그가 채워질 때 RP 비트가 설정된 S,G만 정리합니다. 이를 위해서는 공유 세그먼트에서 다른 데이터 플레인 패킷을 수신해야 합니다. 제어 평면이 (S,G)에 이미 구축되었으므로 공유 세그먼트에서 어설션 메시지를 트리거하는 중복 작업이 발생합니다.