Riflessione servizio multicast con PIM-SM su IOS-XE: Da Multicast a Unicast

Sommario

Introduzione

Lo scopo di questo articolo è quello di fornire una descrizione del funzionamento di base di MSR (Multicast Service Replication) utilizzando piattaforme IOS-XE, attraverso una guida di configurazione.

Prerequisiti

Requisiti

Conoscenze base di PIM-SM

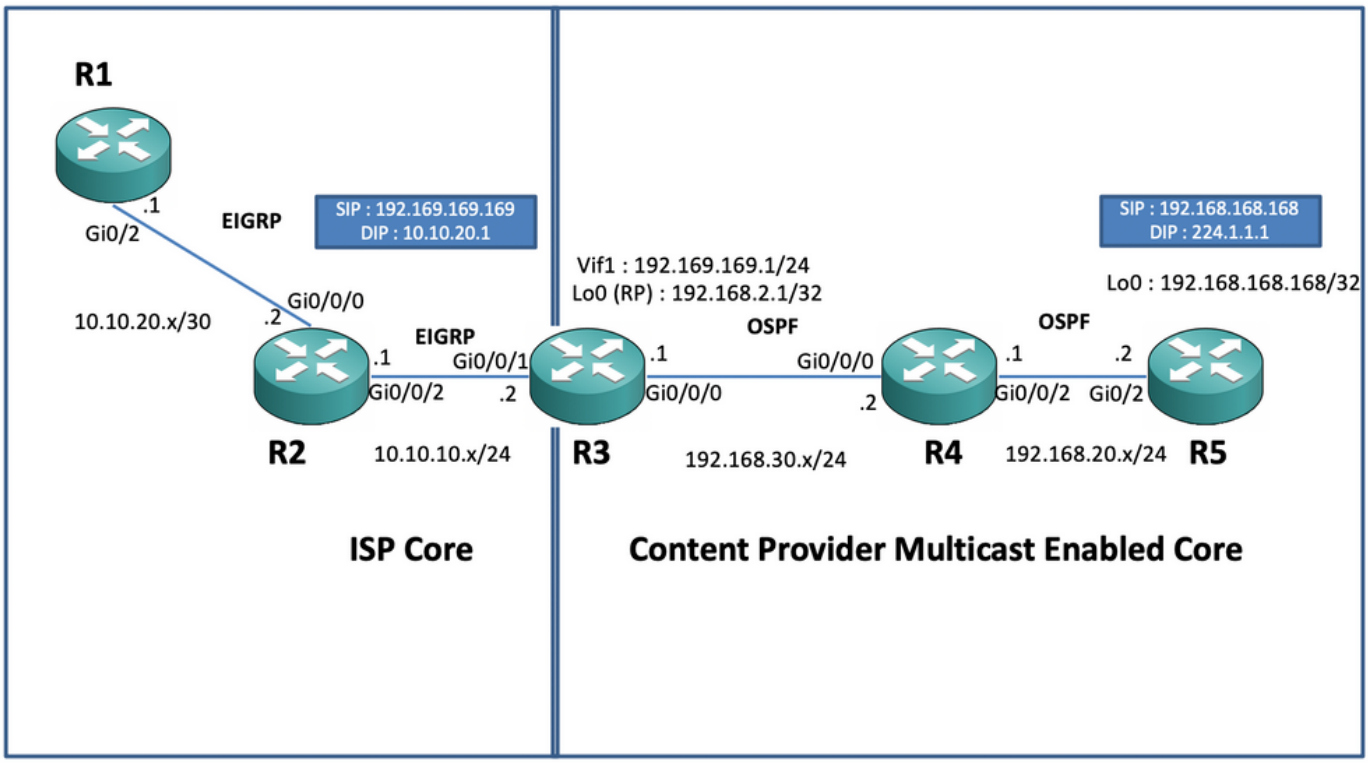
Componenti usati

ASR 1000 (R2&R4), ISR 4300 (R3), ISR 2900 (R1&R5)

Configurazione

Di seguito vengono illustrate le configurazioni end-to-end basate sullo scenario diagrammatico per la traduzione del multicast.

Esempio di rete



Configurazioni

Nel diagramma precedente, il nodo R1 funge da ricevitore che dovrebbe ricevere solo dati multicast unicast dalla sorgente multicast.

Il nodo R5 funge da origine multicast che genera traffico ICMP multicast originato dall'interfaccia 0 di loopback.

Il nodo R2 si trova nel dominio di base multicast dei provider di contenuti ed esegue PIM-SM con underlay di OSPF.

Il nodo R3 funge da router che esegue l'applicazione di replica del servizio multicast e in questo caso è il router di confine multicast dal quale il traffico dati multicast deve essere convertito in un pacchetto dati unicast verso il destinatario. Utilizza OSPF e EIGRP rispettivamente con il Content Provider e ISP e ospita l'RP (Rendezvouz Point) sulla sua interfaccia di loopback nel dominio core multicast.

Il nodo R4 è sotto il controllo dell'ISP Core e non è abilitato per il multicast e comprende solo come raggiungere il nodo R3 utilizzando il routing EIGRP sottostante.

Di seguito sono riportate le configurazioni appropriate presenti sui nodi presenti nel diagramma topologico sopra riportato:

R1:

```
! no ip domain lookup ip cef no ipv6 cef ! interface GigabitEthernet0/2 ip address 10.10.20.1
255.255.255.0 duplex auto speed auto end ! router eigrp 100 network 10.10.20.0 0.0.0.255 !
```

R2:

```
! interface GigabitEthernet0/0/0 ip address 10.10.20.2 255.255.255.0 negotiation auto !
```

```
interface GigabitEthernet0/0/2 ip address 10.10.10.1 255.255.255.0 negotiation auto ! router eigrp 100 network 10.10.10.0 0.0.0.255 network 10.10.20.0 0.0.0.255 !
```

R3:

```
! ip multicast-routing distributed ! interface Loopback0 ip address 192.168.2.1 255.255.255.255 ip pim sparse-mode ip ospf 1 area 0 ! interface GigabitEthernet0/0/0 ip address 192.168.30.1 255.255.255.0 ip pim sparse-mode ip ospf 1 area 0 negotiation auto ! interface GigabitEthernet0/0/1 ip address 10.10.10.2 255.255.255.0 negotiation auto ! interface Vif1 ip address 192.169.169.1 255.255.255.0 ip pim sparse-mode ip service reflect GigabitEthernet0/0/0 destination 224.1.1.0 to 10.10.20.0 mask-len 24 source 192.169.169.169 <<<< ip igmp static-group 224.1.1.1 ip ospf 1 area 0 ! router eigrp 100 network 10.10.10.0 0.0.0.255 ! router ospf 1 ! ip pim rp-address 192.168.2.1 !
```

R4

```
! ip multicast-routing distributed ! interface GigabitEthernet0/0/0 ip address 192.168.30.2 255.255.255.0 ip pim sparse-mode ip ospf 1 area 0 negotiation auto ! interface GigabitEthernet0/0/2 ip address 192.168.20.1 255.255.255.0 ip pim sparse-mode ip ospf 1 area 0 negotiation auto ! router ospf 1 ! ip pim rp-address 192.168.2.1 !
```

R5

```
! ip multicast-routing ip cef no ipv6 cef ! interface Loopback0 ip address 192.168.168.168 255.255.255.255 ip pim sparse-mode ip ospf 1 area 0 ! interface GigabitEthernet0/2 ip address 192.168.20.2 255.255.255.0 ip pim sparse-mode ip ospf 1 area 0 duplex auto speed auto ! router ospf 1 ! ip pim rp-address 192.168.2.1 !
```

Verifica

Èpossibile convalidare le configurazioni eseguendo un ping di prova per simulare il traffico multicast proveniente dal router R5 con un'origine della relativa interfaccia di loopback 0 [192.168.168.168] destinata all'indirizzo multicast 224.1.1.1. Quindi, controllare le voci di route sul nodo che esegue l'applicazione MSR, ad esempio R3:

```
R5(config)#do ping 224.1.1.1 sou lo 0 rep 10000000 Type escape sequence to abort. Sending 10000000, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds: Packet sent with a source address of 192.168.168.168 ..........................
```

```
R3#sh ip mroute 224.1.1.1 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, c - PFP-SA cache created entry Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join Timers: Uptime/Expires Interface state: Interface, Next-Hop or VCD, State/Mode (*, 224.1.1.1), 00:47:41/stopped, RP 192.168.2.1, flags: SJC Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: Vif1, Forward/Sparse, 00:46:36/00:01:23 <<<< (192.168.168.168, 224.1.1.1), 00:00:20/00:02:43, flags: T Incoming interface: GigabitEthernet0/0/0, RPF nbr 192.168.30.2 Outgoing interface list: Vif1, Forward/Sparse, 00:00:20/00:02:39 <<<<
```

```
R3#sh ip mroute 224.1.1.1 count Use "show ip mfib count" to get better response time for a large number of mroutes. IP Multicast Statistics 3 routes using 2938 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: 224.1.1.1,
```

```
Source count: 1, Packets forwarded: 1455, Packets received: 1458 <<<< RP-tree: Forwarding:
1/0/100/0, Other: 1/0/0 Source: 192.168.168.168/32, Forwarding: 1454/1/113/0, Other: 1457/3/0
R3#sh ip mroute 224.1.1.1 count Use "show ip mfib count" to get better response time for a large
number of mroutes. IP Multicast Statistics 3 routes using 2938 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: 224.1.1.1,
Source count: 1, Packets forwarded: 1465, Packets received: 1468 <<<< RP-tree: Forwarding:
1/0/100/0, Other: 1/0/0 Source: 192.168.168.168/32, Forwarding: 1464/1/113/0, Other: 1467/3/0
```

Inoltre, è possibile acquisire le clip per verificare che i pacchetti vengano effettivamente tradotti all'indirizzo di destinazione unicast previsto sul nodo R2 utilizzando la funzione EPC (Embedded Packet Capture) sul router IOS-XE:

```
R2#mon cap TAC int gi 0/0/2 both match any R2#mon cap TAC buff siz 50 circular R2#mon cap TAC
start Started capture point : TAC R2# *Aug 12 06:50:40.195: %BUFCAP-6-ENABLE: Capture Point TAC
enabled. R2#sh mon cap TAC buff br | i ICMP 6 114 10.684022 192.169.169.169 -> 10.10.20.1 0 BE
ICMP <<<< 7 114 10.684022 192.169.169.169 -> 10.10.20.1 0 BE ICMP <<<< 8 114 12.683015
192.169.169.169 -> 10.10.20.1 0 BE ICMP <<<< 9 114 12.683015 192.169.169.169 -> 10.10.20.1 0 BE
ICMP <<<<
```

In questo caso, è importante notare che quando si eseguono regolarmente ping ICMP multicast in "ambienti di emulazione", ci si aspetta di ricevere nuovamente i pacchetti di risposta echo ICMP dal lato del destinatario verso l'origine, supponendo che i due dispositivi (origine e destinatario) siano completamente raggiungibili. Tuttavia, in questo scenario è importante notare che anche se si cerca di pubblicizzare l'indirizzo di origine NATted per i pacchetti ICMP multicast, ad esempio 192.169.169.169 fino al ricevitore, ad esempio da R1 a EIGRP, l'eco ICMP unicast risponde senza attraversare il router R3, poiché il NAT inverso non è configurato sul nodo dell'applicazione MSR. Possiamo verificarlo, cercando di eseguire l'annuncio di route EIGRP dell'interfaccia Vif 1 su R3 in EIGRP (ISP Core routing):

```
ISR4351(config)#router eigrp 100 ISR4351(config-router)#network 192.169.169.0 0.0.0.255 <<<<
```

A questo punto è possibile eseguire il check-in delle clip acquisite sul nodo R2 nelle risposte echo ICMP inviate alla R3:

```
R2#sh mon cap TAC buff br | i ICMP
```

Ma i ping continuerebbero a fallire come mostrato sulla fonte R5:

```
R5(config)#do ping 224.1.1.1 sou lo 0 rep 10000000 Type escape sequence to abort. Sending
10000000, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds: Packet sent with a source
address of 192.168.168.168
.....................................................................
.....................................................................
```

Ora, per ottenere le risposte fino all'origine, è possibile configurare l'inoltro della porta NAT sul nodo dell'applicazione MSR R3 in modo da convertire il traffico destinato a 192.169.169.169 in 192.168.168.168, configurando il protocollo NAT estendibile:

```
R3(config)#int gi 0/0/1 R3(config-if)#ip nat out R3(config-if)#int gi 0/0/0 R3(config-if)#ip nat
ins R3(config-if)#exit R3(config)#ip nat inside source static 192.168.168.168 192.169.169.169
extendable <<<<
```

Controllando ora il nodo R5 di origine, è possibile vedere che la risposta è tornata:

```
R5(config)#do ping 224.1.1.1 sou lo 0 rep 10000000 Type escape sequence to abort. Sending
10000000, 100-byte ICMP Echos to 224.1.1.1, timeout is 2 seconds: Packet sent with a source
```

```
address of 192.168.168.168
.............................................................
```

Quanto sopra è stato appena eseguito per spiegare il flusso del pacchetto e per capire come stabilire il percorso/flusso unicast inverso per il traffico di dati e il traffico multicast a valle. Dal momento che negli scenari di produzione normali, di solito non si incontrano casi/istanze in cui le applicazioni multicast in esecuzione sul lato server/origine richiedono un pacchetto di riconoscimento inverso dai riceventi in un formato unicast.

In base ai test e alle convalide precedenti, avrebbe dovuto fornire una breve panoramica su come eseguire l'applicazione di replica del servizio multicast su uno dei nodi di confine multicast e su come distribuire lo stesso se la stessa procedura illustrata sopra dovesse essere estesa a una distribuzione su larga scala.