

Reflexão de serviço multicast usando PIM-SM no IOS-XE : Multicast para Unicast

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Configurar](#)

[Diagrama de Rede](#)

[Configurações](#)

[Verificar](#)

Introduction

A finalidade deste artigo é dar a você uma compreensão do funcionamento básico do MSR (Multicast Service Replication) usando plataformas IOS-XE, através da forma de um guia de laboratório de configuração.

Prerequisites

Requirements

Compreensão básica do PIM-SM

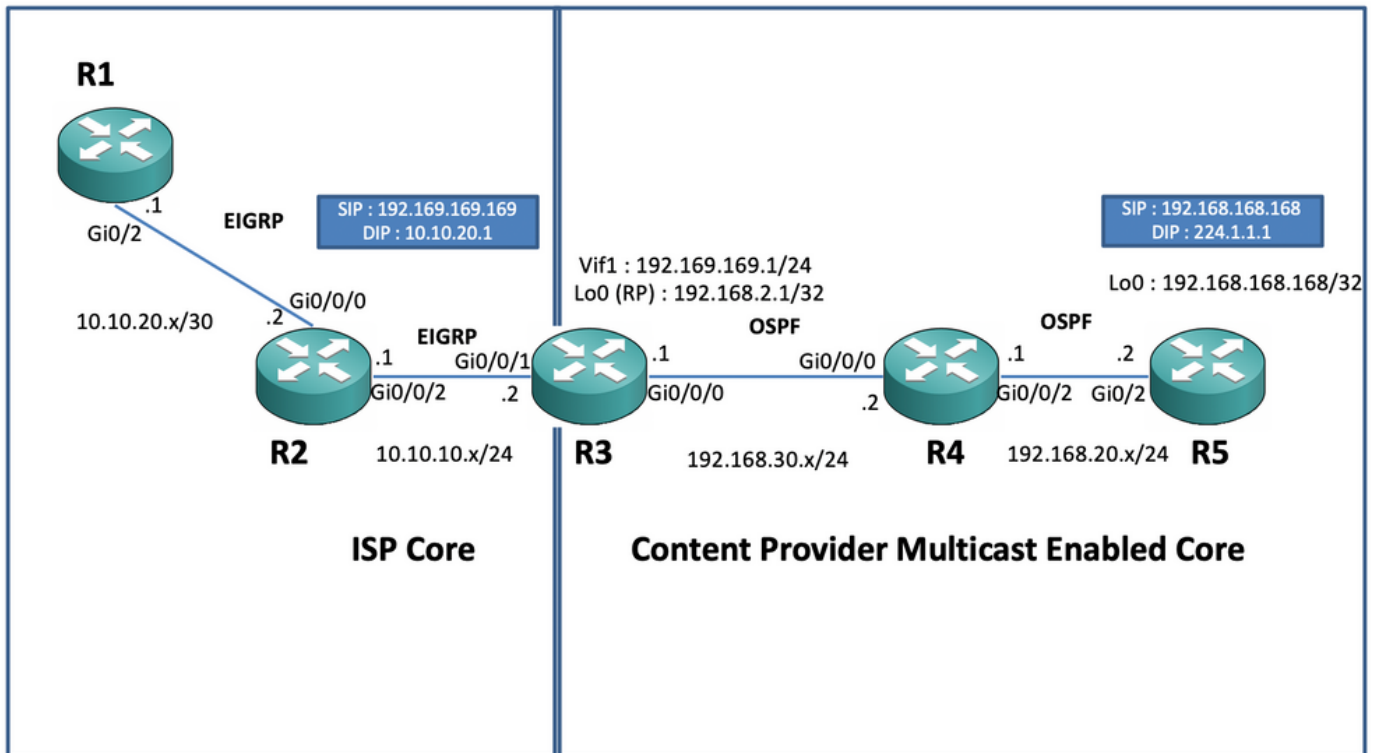
Componentes Utilizados

ASR1000 (R2&R4), ISR4300 (R3), ISR2900 (R1&R5)

Configurar

Mostraríamos abaixo configurações fim-a-fim com base no cenário do diagrama abaixo para a tradução do multicast.

Diagrama de Rede



Configurações

No diagrama acima, o nó R1 atua como o receptor, que deve obter somente feeds de dados multicast unicast da origem multicast.

O nó R5 atua como a origem multicast que gera tráfego ICMP multicast originado de sua interface de loopback 0.

O nó R2 está sob o domínio de núcleo de multicast de provedores de conteúdo e está executando PIM-SM com base de OSPF.

O nó R3 atua como o roteador que executa o Aplicativo de Replicação de Serviço Multicast e é, nesse caso, o roteador de borda de multicast a partir do qual o tráfego de dados multicast deve ser convertido em um pacote de dados unicast para o receptor. Ele usa OSPF e EIGRP com o provedor de conteúdo e ISP respectivamente e hospeda o RP (ponto de rendezvouz) em sua interface de loopback no domínio de núcleo de multicast.

O nó R4 está sob o controle de Núcleo do ISP e não está habilitado para multicast e só entende como acessar o nó R3 usando o roteamento de base do EIGRP.

Abaixo, você pode encontrar as configurações relevantes presentes nos nós presentes no diagrama de topologia acima:

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 !
```

Verificar

Podemos validar as configurações executando um ping de teste para simular o tráfego multicast do roteador R5 com uma origem de sua interface de loopback 0 [192.168.168.168] destinada ao endereço multicast 224.1.1.1. Em seguida, verifique as entradas mroute no nó que está executando o aplicativo MSR, ou seja, 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
```

Além disso, você pode fazer capturas para verificar se os pacotes estão realmente sendo convertidos para o endereço de destino unicast pretendido no nó R2 usando o recurso EPC (Captura de pacote incorporado) no roteador 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 <<<<
```

Aqui, o ponto importante a observar é que regularmente, quando você executa pings ICMP multicast em "ambientes de laboratório", você normalmente espera receber de volta os pacotes de resposta de eco ICMP do lado do receptor para a origem, supondo que haja total acessibilidade entre os dois (origem e receptor). Entretanto, nesse cenário, é importante observar que mesmo que tentemos anunciar o endereço de origem NATted para os pacotes ICMP multicast, ou seja, 192.169.169.169 até o receptor, ou seja, R1 através do EIGRP, ainda assim o eco ICMP unicast responde não cruzará o roteador R3, pois o NAT reverso não está configurado no nó do aplicativo MSR. Podemos testar isso, tentando executar o anúncio de rota EIGRP da interface Vif 1 em R3 no EIGRP (roteamento do Núcleo do ISP) :

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

Agora, podemos verificar as capturas feitas no nó R2 nas respostas de eco ICMP enviadas para R3:

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

Mas os pings ainda falhariam como visto na origem 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
```

Agora, para obter as respostas até a origem, podemos configurar o encaminhamento de portas NAT no nó de aplicação MSR R3 para converter o tráfego destinado para 192.169.169.169 a 192.168.168.168, configurando NAT extensível :

```
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 <<<<
```

Agora, ao verificar o nó R5 de origem, podemos ver a resposta retornar:

```
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

.....
O procedimento acima foi executado apenas para explicar o fluxo do pacote e entender como estabelecer o caminho/fluxo unicast reverso para o tráfego de dados e o tráfego multicast de downstream. Como nos cenários de produção regular, você geralmente não pode vir em casos/instâncias em que os aplicativos multicast em execução no lado servidor/origem exigem um pacote de confirmação reversa dos receptores em uma forma unicast.

Pelos testes e validações acima, ele deve ter fornecido uma breve visão geral sobre como executar o aplicativo de replicação de serviço de multicast em um dos nós de borda de multicast e como implantar o mesmo se o mesmo mostrado acima for expandido para uma implantação em larga escala.