

# Guide de dépannage de multidiffusion IP

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Informations générales](#)

[Le routeur ne transfère pas des paquets multidiffusion à l'hôte en raison d'une panne RPF](#)

[diagnostic du problème](#)

[Correctifs possibles](#)

[Le routeur ne transfère pas des paquets multidiffusion à l'hôte en raison de la configuration TTL de la source](#)

[diagnostic du problème](#)

[Correctifs possibles](#)

[Le routeur ne transfère pas des paquets multidiffusion en raison du seuil de TTL du routeur](#)

[diagnostic du problème](#)

[Correctifs possibles](#)

[Résultat de plusieurs chemins de coût égal dans le comportement non désiré RPF](#)

[diagnostic du problème](#)

[Correctifs possibles](#)

[Pourquoi Multicast IP ne cherche-t-il pas l'équilibre à travers tous les chemins de coût égal disponibles ?](#)

[Correctifs possibles](#)

[Pourquoi recevons-nous des messages d'erreur de Multicast IP « INVALID RP JOIN » sur le routeur ?](#)

[Diagnostiquer le problème - Partie 1](#)

[Correctifs possibles](#)

[Diagnostiquer le problème - Partie 2](#)

[Correctifs possibles](#)

[Le CGMP ne prévient pas la pléthore de paquets de multidiffusion](#)

[diagnostic du problème](#)

[Observations](#)

[Correctifs possibles](#)

[Le CGMP n'empêche pas l'engorgement par paquets de multidiffusion, dû à emplacement de source/récepteur](#)

[diagnostic du problème](#)

[Correctifs possibles](#)

[Le CGMP n'empêche pas l'engorgement par paquets de multidiffusion pour certaines adresses de groupe](#)

[Correctifs possibles](#)

[Des paquets de multidiffusion sont reçus en double](#)

[Cause 1](#)

[Correctif possible 1](#)

[Cause 2](#)

[Correctif possible 2](#)

[Cause 3](#)

[Correctif possible 3](#)

[Pourquoi est-ce que des paquets de multidiffusion sont déposés ?](#)

[Cause 1](#)

[Correctif possible 1](#)

[Cause 2](#)

[Correctif possible 2](#)

[Informations connexes](#)

## Introduction

Ce document décrit des problèmes courants avec leurs solutions pour Multicast IP.

## Conditions préalables

### Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

### [Composants utilisés](#)

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

## [Informations générales](#)

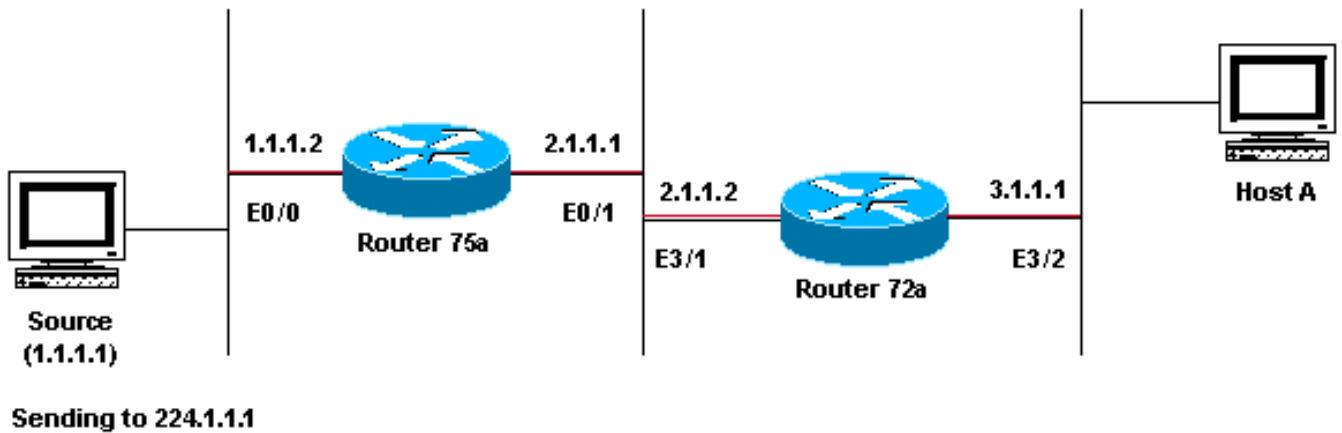
Quand vous dépannez le routage de Multidiffusion, la principale préoccupation est l'adresse source. La Multidiffusion a un concept de contrôle du Reverse Path Forwarding (RPF). Quand un paquet de multidiffusion arrive sur une interface, les contrôles de processus RPF pour s'assurer que cette interface entrante est l'interface sortante les ont utilisé par le routage d'unicast afin d'atteindre la source du paquet de multidiffusion. Ce processus de contrôle RPF empêche des boucles. Le routage de Multidiffusion n'expédie pas un paquet à moins que la source du paquet passe un contrôle RPF. Une fois qu'un paquet aura passé ce contrôle RPF, le routage de multicast transfère le paquet en fonction seulement sur l'adresse de destination.

Comme le routage monodiffusion, le routage multicast a plusieurs protocoles à disposition, tels que le mode dense de Protocol Independent Multicast (PIM-DM), le mode intermédiaire de protocole multicast PIM (PIM-SM), le protocole multicast de routage à vecteur (DVMRP), le protocole BGP multicast (MBGP), et le protocole multicast de découverte de source (MSDP). Les études de cas dans ce document marchent vous par le processus pour dépanner de divers problèmes. Vous verrez quelles commandes sont utilisées afin d'indiquer exactement rapidement le problème et apprendre comment le résoudre. Les études de cas listées ici sont génériques à travers les protocoles, sauf indication contraire.

## [Le routeur ne transfère pas des paquets multidiffusion à l'hôte en](#)

## raison d'une panne RPF

Cette section fournit une solution au problème courant d'une panne du Protocole IP Multicast RPF. Ce schéma de réseau est utilisé comme exemple.



Dans cette figure, les paquets de multidiffusion entrent dans E0/0 du routeur 75a d'un serveur dont l'adresse IP est 1.1.1.1 et envoie pour grouper 224.1.1.1. Ceci est connu comme (S, G) ou (1.1.1.1, 224.1.1.1).

### diagnostic du problème

Les hôtes directement connectés au routeur 75a reçoivent le flux de multicast, mais les hôtes directement connectés au routeur 72a ne le reçoivent pas. D'abord, sélectionnez la commande de **224.1.1.1 de show ip mroute** afin de voir ce qui va en fonction avec le routeur 75a. Cette commande examine l'itinéraire de multidiffusion (mroute) pour l'adresse de groupe 224.1.1.1 :

```
75a#show ip mroute 224.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 00:01:23/00:02:59, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 00:01:23/00:00:00

(1.1.1.1, 224.1.1.1), 00:01:23/00:03:00, flags: TA
  Incoming interface: Ethernet0/0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 00:01:23/00:00:00
```

Puisque le routeur exécute le mode dense PIM (vous savez que c'est mode dense en raison de l'indicateur D), ignorez \*, entrée G et foyez sur le S, entrée G. Cette entrée indique que les paquets de multidiffusion sont originaires d'un serveur dont l'adresse est 1.1.1.1, qui les envoie à un groupe de multidiffusion de 224.1.1.1. Les paquets entrent dans l'interface Ethernet0/0 et sont expédiés l'interface Ethernet0/1. C'est un scénario parfait.

Sélectionnez la commande de **show ip pim neighbor** afin de voir si le routeur 72a affiche le routeur en amont (75a) en tant que voisin PIM :

```
ip22-72a#show ip pim neighbor
PIM Neighbor Table
Neighbor Address  Interface          Uptime    Expires    Ver  Mode
2.1.1.1          Ethernet3/1        2d00h     00:01:15  v2
```

De la sortie de commande de **show ip pim neighbor**, la proximité PIM semble bonne.

Sélectionnez la commande de **show ip mroute** afin de voir si le routeur 72a a le bon mroute :

```
ip22-72a#show 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,
      X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
      U - URD, I - Received Source Specific Host Report, Z - Multicast Tunnel
      Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 00:10:42/stopped, RP 0.0.0.0, flags: DC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet3/1, Forward/Dense, 00:10:42/00:00:00
    Ethernet3/2, Forward/Dense, 00:10:42/00:00:00
```

```
(1.1.1.1, 224.1.1.1), 00:01:10/00:02:48, flags:
  Incoming interface: Ethernet2/0, RPF nbr 0.0.0.0 Outgoing interface list: Ethernet3/1,
Forward/Dense, 00:01:10/00:00:00 Ethernet3/2, Forward/Dense, 00:00:16/00:00:00 ip22-72a#
```

Vous pouvez voir, à l'aide de la commande de **224.1.1.1 de show ip mroute** que l'interface d'entrée est Ethernet2/0, alors qu'Etheret3/1 était attendu.

Sélectionnez la commande de **compte de 224.1.1.1 de show ip mroute** afin de voir si n'importe quel trafic de multidiffusion pour ce groupe arrive au routeur 72a et ce qui se produit ensuite :

```
ip22-72a#show ip mroute 224.1.1.1 count
IP Multicast Statistics
3 routes using 2032 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:      0, Packets received: 471
  Source:      1.1.1.1/32, Forwarding: 0/0/0/0, Other: 471/471/0 ip22-72a#
```

Vous pouvez voir, à l'aide des autres nombres que le trafic est déposé en raison de la panne RPF : un total de 471 abandons en raison de la panne RPF - 471...

Sélectionnez le **<source >** la commande de **show ip rpf** afin de voir s'il y a une erreur RPF :

```
ip22-72a#show ip rpf 1.1.1.1
RPF information for ? (1.1.1.1)
  RPF interface: Ethernet2/0
  RPF neighbor: ? (0.0.0.0)
  RPF route/mask: 1.1.1.1/32
  RPF type: unicast (static)
  RPF recursion count: 0
  Doing distance-preferred lookups across tables
ip22-72a#
```

Cisco IOS® calcule l'interface RPF de cette façon. Les sources possibles d'informations RPF table sont monodiffusé le Tableau de routage, la table de routage MBGP, de DVMRP routage, et table mroute statique. Quand vous calculez l'interface RPF, principalement la distance administrative est utilisée afin de déterminer exactement que la source d'informations le calcul RPF est basé en fonction. Les règles spécifiques sont :

- Toutes les sources précédentes de données RPF sont recherchées pour une correspondance sur l'adresse IP de la source. Quand vous utilisez des arbres partagés, l'adresse RP est utilisée au lieu de l'adresse source.
- Si plus d'une route correspondante est recherchée, la route avec la plus faible distance administrative est utilisée.
- Si les distances administratives sont égales, alors cet ordre de préférence est utilisé :Mroutes statiquesRoutes de DVMRPRoutes MBGPRoutes de monodiffusion
- Si plusieurs entrées se produisent pour une route dans le même tableau de routage, la route de la plus longue correspondance est utilisée.

Le résultat de la commande **show ip rpf 1.1.1.1** montre que l'interface RPF est E2/0, mais l'interface d'entrée devrait être E3/1.

Sélectionnez la commande de **1.1.1.1 de show ip route** afin de voir pourquoi l'interface RPF est différente de ce qu'a été prévu.

```
ip22-72a#show ip route 1.1.1.1
Routing entry for 1.1.1.1/32
Known via "static", distance 1, metric 0 (connected)
Routing Descriptor Blocks:
* directly connected, via Ethernet2/0
Route metric is 0, traffic share count is 1
```

Vous pouvez voir de ce résultat de la commande **montrer ip route 1.1.1.1** qu'il y a une route statique de /32, qui fait que l'interface fautive est choisie comme interface RPF.

Sélectionnez quelques autres commandes de **débogage** :

```
ip22-72a#debug ip mpacket 224.1.1.1
*Jan 14 09:45:32.972: IP: s=1.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
*Jan 14 09:45:33.020: IP: s=1.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
*Jan 14 09:45:33.072: IP: s=1.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
*Jan 14 09:45:33.120: IP: s=1.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
```

Les paquets entrés sur E3/1, qui est correct. Cependant, ils sont lâchés parce que ce n'est pas l'interface que la table de routage d'unicast l'utilise pour le contrôle RPF.

Remarque: Déboguer des paquets est dangereux. L'élimination des imperfections de paquet déclenche la commutation de processus des paquets de multidiffusion, qui est CPU intensive. En outre, le débogage de paquets peut produire un résultat énorme et encombrer le routeur complètement devant parce que la sortie vers le port de console est lente. Avant que vous des debugs packets, soin particulier deviez être commande rentrée pour désactiver la sortie de journalisation à la console, et à l'enable se connectant au tampon mémoire. Afin de réaliser ceci, configurez le **no logging console** et le **logging buffered debugging**. Les résultats du débogage peuvent être consultés avec la commande de **show logging**.

## Correctifs possibles

Vous pouvez ou changer la table de routage d'unicast afin de répondre à cette exigence ou vous pouvez ajouter un mroute statique pour expulser la Multidiffusion au RPF une interface spécifique, indépendamment de ce que la table de routage d'unicast énonce. Ajouter un mroute statique :

```
ip22-72a(config)#ip mroute 1.1.1.1 255.255.255.255 2.1.1.1
```

Ce mroute de charge statique déclare cela pour obtenir à l'adresse 1.1.1.1 pour le RPF, utilisent 2.1.1.1 comme prochain saut qui est l'interface E3/1.

```
ip22-72a#show ip rpf 1.1.1.1
RPF information for ? (1.1.1.1)
  RPF interface: Ethernet3/1
  RPF neighbor: ? (2.1.1.1)
  RPF route/mask: 1.1.1.1/32
  RPF type: static mroute
  RPF recursion count: 0
  Doing distance-preferred lookups across tables
```

La sortie du **show ip mroute** et du **debug ip mpacket** semble bonne, le nombre de paquets envoyés dans le **compte de show ip mroute** augmente, et le HostA reçoit des paquets.

```
ip22-72a#show ip mroute 224.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 00:01:15/00:02:59, RP 0.0.0.0, flags: DJC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet3/1, Forward/Sparse-Dense, 00:01:15/00:00:00
    Ethernet3/2, Forward/Sparse-Dense, 00:00:58/00:00:00

(1.1.1.1, 224.1.1.1), 00:00:48/00:02:59, flags: CTA
  Incoming interface: Ethernet3/1, RPF nbr 2.1.1.1, Mroute
  Outgoing interface list:
    Ethernet3/2, Forward/Sparse-Dense, 00:00:48/00:00:00
```

```
ip22-72a#show ip mroute 224.1.1.1 count
IP Multicast Statistics
3 routes using 2378 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: 1019, Packets received: 1019
  Source: 1.1.1.1/32, Forwarding: 1019/1/100/0, Other: 1019/0/0
```

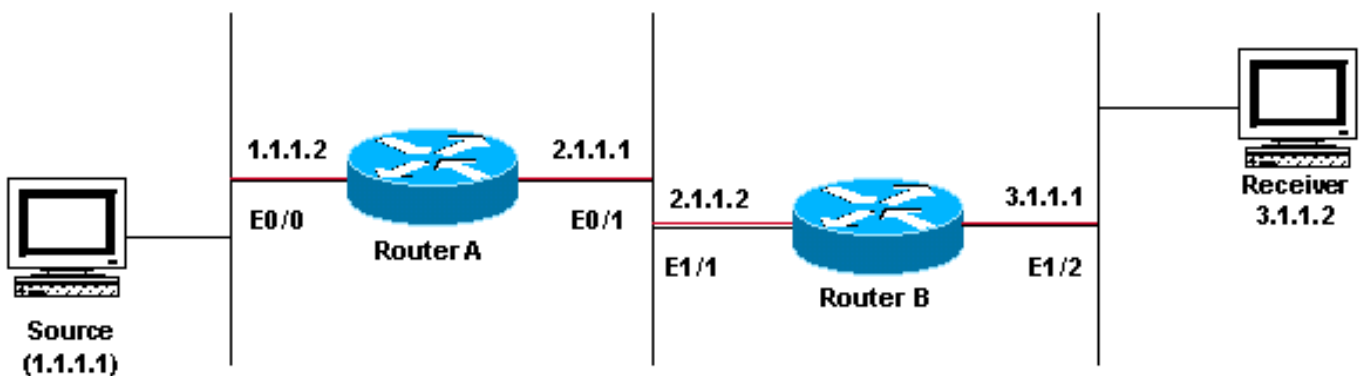
```
ip22-72a#show ip mroute 224.1.1.1 count
IP Multicast Statistics
3 routes using 2378 bytes of memory
2 groups, 0.50 average sources per group
Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)
```

```
Group: 224.1.1.1, Source count: 1, Packets forwarded: 1026, Packets received: 1026
  Source: 1.1.1.1/32, Forwarding: 1026/1/100/0, Other: 1026/0/0
ip22-72a#
```

```
ip22-72a#debug ip mpacket 224.1.1.1
*Jan 14 10:18:29.951: IP: s=1.1.1.1 (Ethernet3/1)
d=224.1.1.1 (Ethernet3/2) len 60, mforward
*Jan 14 10:18:29.999: IP: s=1.1.1.1 (Ethernet3/1)
d=224.1.1.1 (Ethernet3/2) len 60, mforward
*Jan 14 10:18:30.051: IP: s=1.1.1.1 (Ethernet3/1)
d=224.1.1.1 (Ethernet3/2) len 60, mforward
```

## Le routeur ne transfère pas des paquets multidiffusion à l'hôte en raison de la configuration TTL de la source

Cette section fournit une solution au problème courant des paquets de Protocole IP Multicast qui ne sont pas expédiés parce que la valeur du Time to Live (TTL) est décrétementée à zéro. C'est un problème courant, car il y a beaucoup d'applications multicast. Ces applications multidiffusion sont conçues principalement pour l'utilisation de LAN, et définissent ainsi le TTL à une valeur basse ou même à un 1. Ce schéma de réseau est utilisé comme exemple.



**Sending to 224.1.1.1**

## diagnostic du problème

Dans la figure précédente, le routeur A n'expédie pas des paquets des sources au récepteur directement connecté du routeur B. Regardez la sortie de la commande de **show ip mroute** sur le routeur A afin de voir l'état de routage de Multidiffusion :

```
ROUTERA#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 00:00:01/00:00:00, RP 0.0.0.0, flags: DJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 00:00:01/00:00:00

(*, 224.1.1.1), 00:00:02/00:02:57, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
```

Outgoing interface list:

Ethernet0/1, Forward/Sparse-Dense, 00:00:02/00:00:00

Vous pouvez ignorer 224.0.1.40 dans la sortie puisque tous les Routeurs joignent ce groupe de détection d'auto-RP. Mais il n'y a aucune source listée pour 224.1.1.1. En plus de « \*, 224.1.1.1 » vous devriez consulter "1.1.1.1, 224.1.1.1".

Sélectionnez la commande de **show ip rpf** afin de voir si c'est une question RPF :

```
ROUTERA#show ip rpf 1.1.1.1
RPF information for ? (1.1.1.1)
  RPF interface: Ethernet0/0
  RPF neighbor: ? (0.0.0.0) - directly connected
  RPF route/mask: 1.1.1.0/24
  RPF type: unicast (connected)
  RPF recursion count: 0
  Doing distance-preferred lookups across tables
```

De la sortie, vous voyez que ce n'est pas une question RPF. Le contrôle RPF précise correctement E0/0 pour atteindre l'adresse IP de la source.

Vérifiez si PIM est configuré sur les interfaces avec la commande de **show ip pim interface** :

```
ROUTERA#show ip pim interface
```

Address	Interface	Version/Mode	Nbr Count	Query Intvl	DR
1.1.1.2	Ethernet0/0	v2/Sparse-Dense	0	30	1.1.1.2
2.1.1.1	Ethernet0/1	v2/Sparse-Dense	2	30	2.1.1.2

Le résultat semble bon, donc ce n'est pas le problème. Vérifiez si le routeur identifie n'importe quel trafic actif avec la commande **active de show ip mroute** :

```
ROUTERA#show ip mroute active
Active IP Multicast Sources - sending >= 4 kbps
```

Basé sur la sortie, le routeur n'identifie aucun trafic actif.

```
ROUTERA#debug ip mpacket
IP multicast packets debugging is on
```

Peut-être que le récepteur n'envoie aucun rapport de protocole de gestion de groupe Internet (IGMP) (relié) pour le groupe 224.1.1.1. Vous pouvez le vérifier avec la commande de **groupe d'igmp de show ip** :

```
ROUTERB#show ip igmp group
IGMP Connected Group Membership
Group Address  Interface      Uptime    Expires    Last Reporter
224.0.1.40     Ethernet1/1    00:34:34  never      2.1.1.2
224.1.1.1      Ethernet1/2    00:30:02  00:02:45  3.1.1.2
```

224.1.1.1 a été connecté à E1/2, ce qui est en ordre.

Le mode dense du protocole multicast PIM est un protocole inondation et élagage, il n'y a donc pas de liaisons, mais il y a des greffes. Puisque le routeur B n'a pas été inondé par le routeur A, il ne sait pas où envoyer une greffe.

Vous pouvez vérifier si c'est un problème de TTL avec la saisie d'analyseur et également avec la commande de **show ip traffic** :

```
ROUTERA#show ip traffic
IP statistics:
  Rcvd: 248756 total, 1185 local destination
```



```
0 format errors, 0 checksum errors, 63744 bad hop count
0 unknown protocol, 0 not a gateway
0 security failures, 0 bad options, 0 with options
```

La sortie affiche 63744 mauvais comptes de saut. Chaque fois que vous introduisez cette commande, le mauvais nombre de sauts augmente. C'est une indication forte que l'expéditeur envoie à des paquets avec un TTL=1, que le routeur A décrémente à zéro. Ceci a comme conséquence une augmentation de la mauvaise valeur du nombre de sauts.

## Correctifs possibles

Afin de résoudre le problème, vous devez augmenter le TTL. Ceci est fait au niveau de l'application sur l'expéditeur. Pour plus d'informations, consultez votre manuel d'instructions de l'application multidiffusion.

Une fois que ceci est fait, le routeur A ressemble à ceci :

```
ROUTERA#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 01:16:32/00:00:00, RP 0.0.0.0, flags: DJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 01:16:32/00:00:00

(*, 224.1.1.1), 00:28:42/00:02:59, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 00:28:42/00:00:00

(1.1.1.1, 224.1.1.1), 00:19:24/00:02:59, flags: TA
  Incoming interface: Ethernet0/0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 00:19:24/00:00:00
```

C'est le résultat que vous voulez voir.

Sur le routeur B vous voyez :

```
ROUTERB#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.0.1.40), 01:23:57/00:00:00, RP 0.0.0.0, flags: DJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet1/1, Forward/Sparse-Dense, 01:23:57/00:00:00

(*, 224.1.1.1), 01:19:26/00:02:59, RP 0.0.0.0, flags: DJC
  Incoming interface: Null, RPF nbr 0.0.0.0
```

```
Outgoing interface list:
 Ethernet1/1, Forward/Sparse-Dense, 01:19:26/00:00:00
 Ethernet1/2, Forward/Sparse-Dense, 01:19:26/00:00:00
```

```
(1.1.1.1, 224.1.1.1), 00:17:46/00:02:59, flags: CTA
```

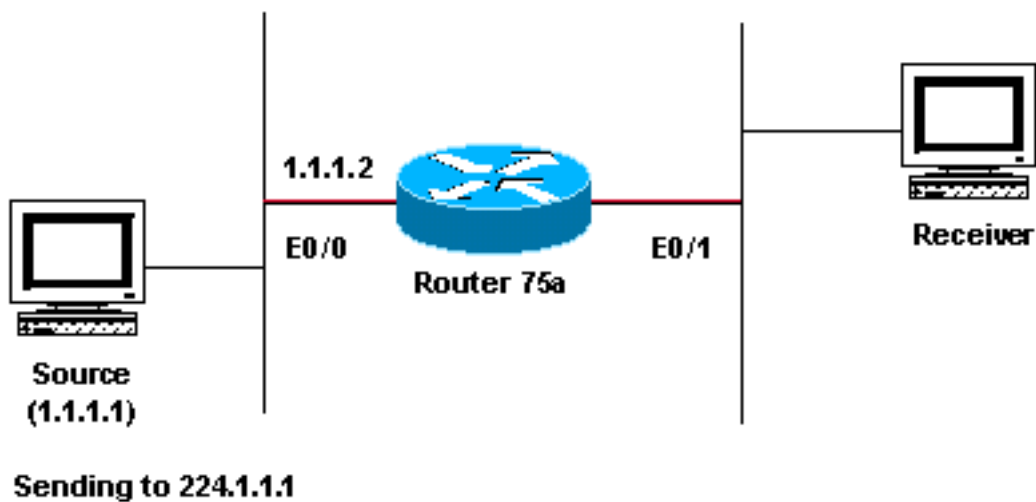
```
Incoming interface: Ethernet1/1, RPF nbr 2.1.1.1
```

```
Outgoing interface list:
```

```
 Ethernet1/2, Forward/Sparse-Dense, 00:17:46/00:00:00
```

## Le routeur ne transfère pas des paquets multidiffusion en raison du seuil de TTL du routeur

Cette section fournit une solution au problème courant où le seuil TTL est placé si bas, de sorte que le trafic de Protocole IP Multicast n'atteigne pas le récepteur. Ce schéma de réseau est utilisé comme exemple.



### diagnostic du problème

Dans la figure précédente, le récepteur ne reçoit pas des paquets de multidiffusion de la source. Il pourrait y avoir plusieurs Routeurs entre la source et le routeur 75a. Observez d'abord le routeur 75a, puisqu'il est directement connecté au récepteur.

```
ip22-75a#show ip mroute 224.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 00:32:05/00:02:59, RP 0.0.0.0, flags: DJC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 00:08:17/00:00:00

(1.1.1.1, 224.1.1.1), 00:01:02/00:01:57, flags: CTA
  Incoming interface: Ethernet0/0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/1, Forward/Sparse-Dense, 00:01:02/00:00:00
```

La sortie affiche les paquets Ethernet0/1 du routeur 75a en avant. Afin d'être le routeur

absolument sûr 75a en avant les paquets, activez **mettent au point** juste pour ces source et groupe de multidiffusion :

```
ip22-75a#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ip22-75a(config)#access-list 101 permit udp host 1.1.1.1 host 224.1.1.1
ip22-75a(config)#end
ip22-75a#debug ip mpacket 101
IP multicast packets debugging is on for access list 101
ip22-75a#
*Jan 17 09:04:08.714: IP: s=1.1.1.1 (Ethernet0/0) d=224.1.1.1 len 60, threshold denied
*Jan 17 09:04:08.762: IP: s=1.1.1.1 (Ethernet0/0) d=224.1.1.1 len 60, threshold denied
*Jan 17 09:04:08.814: IP: s=1.1.1.1 (Ethernet0/0) d=224.1.1.1 len 60, threshold denied
```

**Les messages de débogage** indiquent que le routeur 75a n'expédie pas les paquets de multidiffusion parce que le seuil TTL a été atteint. Regardez la configuration de routeur afin de voir si vous pouvez trouver la raison. Ce résultat montre le coupable :

```
interface Ethernet0/1
 ip address 2.1.1.1 255.255.255.0
 ip pim sparse-dense-mode
 ip multicast ttl-threshold 15
```

Le routeur a un seuil de TTL de 15, mais ceci ne signifie pas que quelque chose plus grand qu'un TTL de 15 n'est pas envoyé. En fait, l'opposé est vrai. L'application est envoyée avec un TTL de 15. Quand qu'elle arrive au routeur 75a, les paquets de multidiffusion ont un TTL de moins de 15.

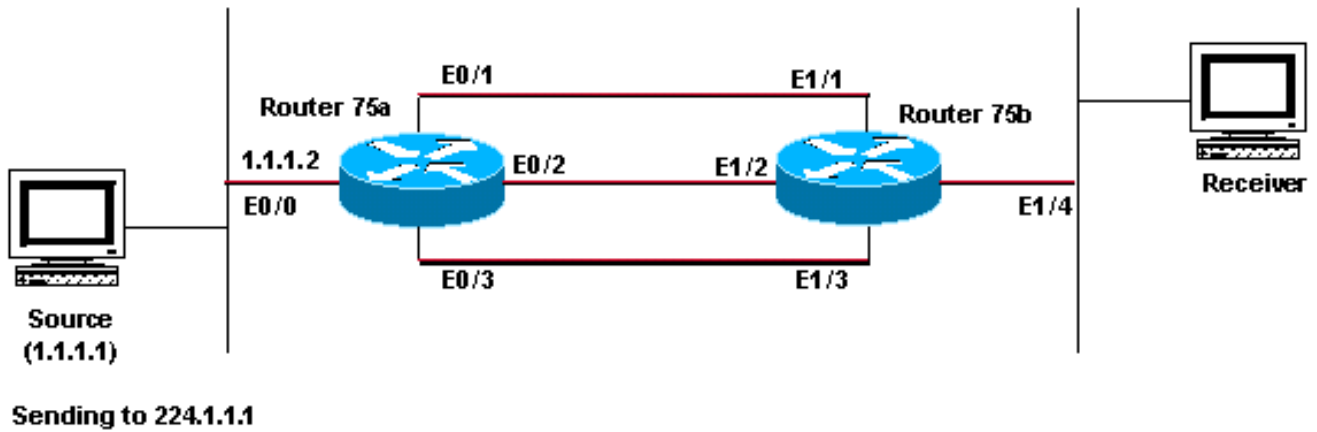
La commande **ip multicast ttl-threshold <value>** signifie que les paquets avec un TTL inférieur au seuil spécifié, dans ce cas, 15, ne sont pas transférée. Cette commande est habituellement utilisée afin de fournir un cadre pour garder le trafic de multidiffusion interne de la dérive hors de l'intranet.

## Correctifs possibles

L'un ou l'autre retire le **<value >** la commande de **TTL-seuil de Protocole IP Multicast** avec le **forme no de** cette commande, qui retourne à la valeur seuil du par défaut TTL de 0, ou diminue le seuil TTL de sorte que le trafic puisse passer.

## Résultat de plusieurs chemins de coût égal dans le comportement non désiré RPF

Cette section affiche comment les chemins de coût égal à une source multicast peuvent entraîner le comportement non désiré RPF. Il décrit également comment configurer le Protocole IP Multicast afin d'éviter ce comportement. Ce schéma de réseau est utilisé comme exemple.



## diagnostic du problème

Dans la figure, le routeur 75b a trois chemins de coût égal de nouveau à la source (1.1.1.1), et elle choisit un lien qui vous ne voulez pas que soit son premier choix comme lien RPF.

Confronté aux plusieurs chemins à une source de coût égal, multicast IP choisit l'interface qui a un voisin de multidiffusion indépendant du protocole (PIM) avec l'adresse IP la plus élevée comme interface d'entrée et puis des éléments aux voisins PIM sur les autres liaisons.

## Correctifs possibles

Afin de changer le Protocole IP Multicast d'interface choisit en tant que son interface entrante, vous peut faire un de ces derniers :

- Configurer seulement PIM sur les interfaces que vous voulez que le multicast traverse, ce qui signifie que vous perdez la redondance de multicast.
- Modifier les sous-réseaux de sorte que l'adresse IP la plus élevée se trouve sur la liaison que vous voulez avoir comme liaison multicast primaire.
- Créez une route multicast statique (mroute) qui précise l'interface préférée de Multidiffusion, qui signifie que vous perdez la Redondance de Multidiffusion.

Comme exemple, un mroute statique est créé.

Avant que vous installiez un mroute statique, vous voyez dans cette sortie que la table de routage a trois artères de coût égal pour l'adresse source 1.1.1.1. Les informations RPF indiquent que l'interface RPF est E1/3 :

```
ip22-75b#show ip route 1.1.1.1
Routing entry for 1.1.1.0/24
  Known via "ospf 1", distance 110, metric 20, type intra area
  Redistributing via ospf 1
  Last update from 3.1.1.1 on Ethernet1/2, 00:26:21 ago
  Routing Descriptor Blocks:
  * 4.1.1.1, from 10.0.119.66, 00:26:21 ago, via Ethernet1/3
    Route metric is 20, traffic share count is 1
  2.1.1.1, from 10.0.119.66, 00:26:21 ago, via Ethernet1/1
    Route metric is 20, traffic share count is 1
  3.1.1.1, from 10.0.119.66, 00:26:21 ago, via Ethernet1/2
    Route metric is 20, traffic share count is 1
```

```
ip22-75b#show ip rpf 1.1.1.1
```

```

RPF information for ? (1.1.1.1)
  RPF interface: Ethernet1/3
  RPF neighbor: ? (4.1.1.1)
  RPF route/mask: 1.1.1.0/24
  RPF type: unicast (ospf 1)
  RPF recursion count: 0
  Doing distance-preferred lookups across tables

ip22-75b#show ip mroute 224.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 01:30:34/00:02:59, RP 0.0.0.0, flags: DJC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet1/4, Forward/Sparse-Dense, 01:30:34/00:00:00
    Ethernet1/1, Forward/Sparse-Dense, 01:30:35/00:00:00
    Ethernet1/2, Forward/Sparse-Dense, 01:30:35/00:00:00
    Ethernet1/3, Forward/Sparse-Dense, 00:24:22/00:00:00

(1.1.1.1, 224.1.1.1), 01:30:35/00:02:59, flags: CT
  Incoming interface: Ethernet1/3, RPF nbr 4.1.1.1
  Outgoing interface list:
    Ethernet1/1, Prune/Sparse-Dense, 01:30:35/00:02:32
    Ethernet1/4, Forward/Sparse-Dense, 01:30:35/00:00:00
    Ethernet1/2, Prune/Sparse-Dense, 00:24:22/00:02:42

```

**Après que vous configuriez le mroute statique, vous voyez en cela sorti l'interface RPF a changé à E1/1 :**

```

ip22-75b#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ip22-75b(config)#ip mroute 0.0.0.0 0.0.0.0 2.1.1.1
ip22-75b(config)#end

ip22-75b#show ip rpf 1.1.1.1
RPF information for ? (1.1.1.1)
  RPF interface: Ethernet1/1
  RPF neighbor: ? (2.1.1.1)
  RPF route/mask: 1.1.1.1/0
  RPF type: static mroute
  RPF recursion count: 0
  Doing distance-preferred lookups across tables

ip22-75b#show ip route 1.1.1.1
Routing entry for 1.1.1.0/24
  Known via "ospf 1", distance 110, metric 20, type intra area
  Redistributing via ospf 1
  Last update from 3.1.1.1 on Ethernet1/2, 00:26:21 ago
  Routing Descriptor Blocks:
  * 4.1.1.1, from 10.0.119.66, 00:26:21 ago, via Ethernet1/3
    Route metric is 20, traffic share count is 1
    2.1.1.1, from 10.0.119.66, 00:26:21 ago, via Ethernet1/1
    Route metric is 20, traffic share count is 1
    3.1.1.1, from 10.0.119.66, 00:26:21 ago, via Ethernet1/2
    Route metric is 20, traffic share count is 1

ip22-75b#show ip mroute 224.1.1.1
IP Multicast Routing Table

```

Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned  
 R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT  
 M - MSDP created entry, X - Proxy Join Timer Running  
 A - Advertised via MSDP

Timers: Uptime/Expires

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

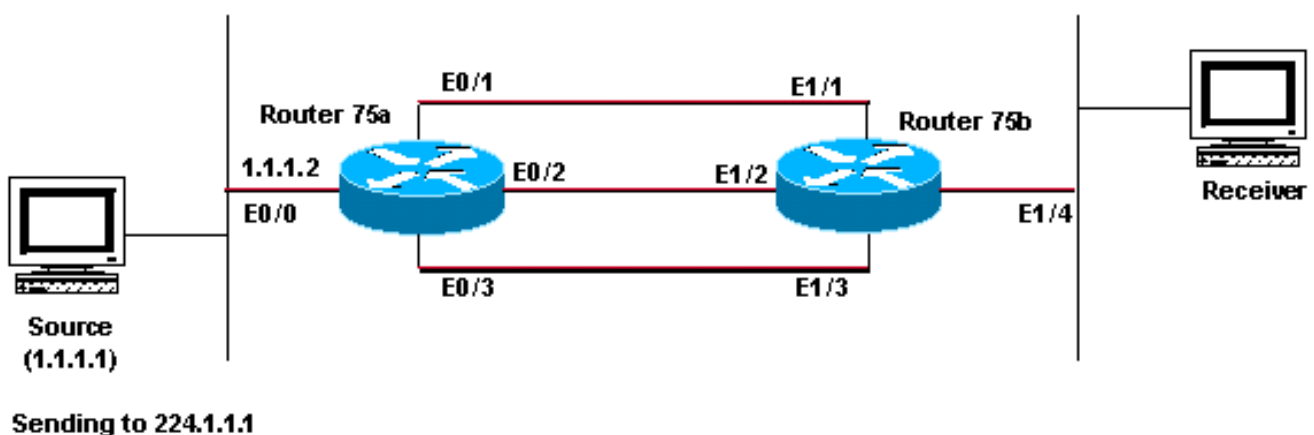
```
(*, 224.1.1.1), 01:31:29/00:02:59, RP 0.0.0.0, flags: DJC
Incoming interface: Null, RPF nbr 0.0.0.0
Outgoing interface list:
Ethernet1/4, Forward/Sparse-Dense, 01:31:29/00:00:00
Ethernet1/1, Forward/Sparse-Dense, 01:31:30/00:00:00
Ethernet1/2, Forward/Sparse-Dense, 01:31:30/00:00:00
Ethernet1/3, Forward/Sparse-Dense, 00:25:17/00:00:00
```

```
(1.1.1.1, 224.1.1.1), 01:31:30/00:02:59, flags: CT
Incoming interface: Ethernet1/1, RPF nbr 2.1.1.1, Mroute
Outgoing interface list:
Ethernet1/4, Forward/Sparse-Dense, 01:31:30/00:00:00
Ethernet1/2, Prune/Sparse-Dense, 00:25:17/00:01:47
Ethernet1/3, Prune/Sparse-Dense, 00:00:31/00:02:28
```

## Pourquoi Multicast IP ne cherche-t-il pas l'équilibre à travers tous les chemins de coût égal disponibles ?

Cette section fournit une solution au problème courant de la façon configurer le Protocole IP Multicast afin d'équilibrer la charge à travers tous les chemins disponibles de coût égal. Ce schéma de réseau est utilisé comme exemple.

Remarque: Avant que vous chargiez le trafic fendu de Protocole IP Multicast à travers des chemins de coût égal au-dessus d'un tunnel, configurez l'équilibrage de charge par paquet de CEF ou bien les paquets GRE ne seront pas chargement équilibré par paquet. Pour que d'autres méthodes chargent le partage dans des environnements de Multidiffusion, voir le [chargement séparer le Protocole IP Multicast pour trafiquer au-dessus d'ECMP](#).



Dans la figure, le routeur 75b a trois chemins de coût égal de nouveau à la source (1.1.1.1). Vous voulez équilibrer le trafic de multidiffusion à travers chacune des trois liaisons.

### Correctifs possibles

Comme vous avez vu dans les [plusieurs chemins de coût égal pour avoir comme conséquence la](#)

section [non désirée de comportement RPF](#), le Protocol Independent Multicast (PIM) choisit seulement une interface pour le contrôle RPF et taille les autres. Ceci signifie que l'équilibrage de charge ne se produit pas. Pour équilibrer la charge, vous devez supprimer PIM des liaisons redondantes et créer un tunnel entre le routeur 75a et le routeur 75b. Vous pouvez alors équilibrer la charge au niveau de liaison, et l'IP s'exécute via le tunnel.

Ce sont les configurations pour le tunnel.

### Routeur 75a

```
interface Tunnel0
 ip address 6.1.1.1 255.255.255.0
 ip pim sparse-dense-mode
 tunnel source Ethernet0/0
 tunnel destination 5.1.1.1
!
interface Ethernet0/0
 ip address 1.1.1.2 255.255.255.0
 ip pim sparse-dense-mode
!
interface Ethernet0/1
 ip address 2.1.1.1 255.255.255.0
!
interface Ethernet0/2
 ip address 3.1.1.1 255.255.255.0
!
interface Ethernet0/3
 ip address 4.1.1.1 255.255.255.0
```

### Routeur 75b

```
interface Tunnel0
 ip address 6.1.1.2 255.255.255.0
 ip pim sparse-dense-mode
 tunnel source Ethernet1/4
 tunnel destination 1.1.1.2
!
interface Ethernet1/1
 ip address 2.1.1.2 255.255.255.0
!
interface Ethernet1/2
 ip address 3.1.1.2 255.255.255.0
!
interface Ethernet1/3
 ip address 4.1.1.2 255.255.255.0
!
interface Ethernet1/4
 ip address 5.1.1.1 255.255.255.0
 ip pim sparse-dense-mode
!
ip mroute 0.0.0.0 0.0.0.0 Tunnel0
```

Après que vous configureriez le tunnel, sélectionnez la commande de **show ip mroute** afin de voir la route multicast (mroute) pour le groupe :

```
ip22-75a#show ip mroute 224.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode
```

```
(* , 224.1.1.1), 02:40:31/00:02:59, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel0, Forward/Sparse-Dense, 00:20:55/00:00:00
```

```
(1.1.1.1, 224.1.1.1), 02:40:32/00:03:29, flags: TA
  Incoming interface: Ethernet0/0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel0, Forward/Sparse-Dense, 00:20:55/00:00:00
```

```
ip22-75b#show ip mroute 224.1.1.1
```

```
IP Multicast Routing Table
```

```
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
```

```
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
```

```
       M - MSDP created entry, X - Proxy Join Timer Running
```

```
       A - Advertised via MSDP
```

```
Timers: Uptime/Expires
```

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

```
(* , 224.1.1.1), 02:42:20/00:02:59, RP 0.0.0.0, flags: DJC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Tunnel0, Forward/Sparse-Dense, 00:22:53/00:00:00
    Ethernet1/4, Forward/Sparse-Dense, 02:42:20/00:00:00
```

```
(1.1.1.1, 224.1.1.1), 00:22:53/00:02:59, flags: CT
  Incoming interface: Tunnel0, RPF nbr 6.1.1.1, Mroute
  Outgoing interface list:
    Ethernet1/4, Forward/Sparse-Dense, 00:22:53/00:00:00
```

Afin de vérifier que les données multicast sont le chargement a équilibré également à travers les trois liens, regardent les données d'interface du routeur 75a.

L'interface d'entrée a 9000 octets/sec :

```
ip22-75a#show interface e0/0
```

```
.
```

```
.
```

```
 5 minute input rate 9000 bits/sec, 20 packets/sec
```

Les trois interfaces de sortie montrent chacune 3000 octets/sec :

```
ip22-75a#show interface e0/1
```

```
.
```

```
.
```

```
 5 minute output rate 3000 bits/sec, 7 packets/sec
```

```
ip22-75a#show interface e0/2
```

```
.
```

```
.
```

```
 5 minute output rate 3000 bits/sec, 7 packets/sec
```

```
ip22-75a#show interface e0/3
```

```
.
```

```
.
```

```
 5 minute output rate 3000 bits/sec, 7 packets/sec
```

**[Pourquoi recevons-nous des messages d'erreur de Multicast IP « INVALID\\_RP\\_JOIN » sur le routeur ?](#)**



Cette section fournit des solutions au problème courant du message d'erreur de Multicast IP du « INVALID\_RP\_JOIN ».

## [Diagnostiquer le problème - Partie 1](#)

Ce message d'erreur est reçu sur le point de rendez-vous (RP) :

```
00:55:15: %PIM-6-INVALID_RP_JOIN: Received (*, 224.1.1.1)
Join from 1.1.6.2 for invalid RP 1.1.5.4
00:56:15: %PIM-6-INVALID_RP_JOIN: Received (*, 224.1.1.1)
Join from 1.1.6.2 for invalid RP 1.1.5.4
```

[Les messages d'erreur de système du logiciel Cisco IOS](#) expliquent ce qui cause cette erreur : un routeur PIM en aval a envoyé un message de connexion pour l'arborescence partagée que ce routeur ne veut pas accepter. Ce comportement indique que ce routeur permet seulement à des routeurs en aval de se connecter à un RP spécifique.

On peut imaginer qu'un certain type de filtrage est en place. Vous devez examiner la configuration du routeur :

```
interface Ethernet0/0
 ip address 1.1.5.4 255.255.255.0
 ip pim sparse-dense-mode
!
ip pim accept-rp 10.2.2.2 8
ip pim send-rp-announce Ethernet0/0 scope 15
ip pim send-rp-discovery scope 15
!
access-list 8 permit 224.0.0.0 0.255.255.255
```

Quel est l'instruction **accept-rp** en configuration censée accomplir ? Dans [Commandes de routage Multicast IP](#), ce document déclare que « pour configurer un routeur pour accepter des connexions ou éléments destinés à un RP spécifié et à une liste spécifique de groupes, il faudra utiliser la commande de configuration globale d'**ip pim accept-rp**. Pour supprimer ce contrôle, n'utilisez l'**aucune** forme de cette commande. »

Quand vous supprimez la commande **ip pim accept-rp**, les messages disparaissent. La commande qui pose le problème a été trouvée, mais ce qui si vous voulez maintenir cette commande dans la configuration ? Vous pourriez permettre l'adresse fausse RP. Sélectionnez la commande de **show ip pim rp mapping** afin de voir l'adresse correcte RP :

```
ip22-75a#show ip pim rp mapping
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)
This system is an RP-mapping agent

Group(s) 224.0.0.0/4
  RP 1.1.5.4 (?), v2v1
    Info source: 1.1.5.4 (?), via Auto-RP
    Uptime: 01:00:48, expires: 00:02:07
```

Selon la sortie, 1.1.5.4 est le seul RP appris par l'intermédiaire de l'auto-RP ou autrement. Cependant, ce routeur est seulement le RP pour les groupes 224.0.0.0/4. Ainsi, l'instruction **accept-rp de pim** dans la configuration est erroné.

## [Correctifs possibles](#)

La solution est de corriger l'adresse IP dans l'instruction d'**ip pim accept-rp** comme suit :

## Modifiez cette instruction

```
ip pim accept-rp 10.2.2.2 8
```

À ceci :

```
ip pim accept-rp 1.1.5.4 8
```

Vous pouvez également changer la déclaration pour recevoir ce qui est dans le cache d'auto-RP, et pour s'assurer que la liste d'accès (8 dans cet exemple) permet la plage de groupe multicast nécessaire. Voici un exemple :

```
ip pim accept-rp auto-rp
```

```
access-list 8 permit 224.0.0.0 0.255.255.255
```

## [Diagnostiquer le problème - Partie 2](#)

Ce message d'erreur est visible sur le router2.

```
router2#
*Aug 12 15:18:17.891:
  %PIM-6-INVALID_RP_JOIN: Received (*, 224.0.1.40)
  Join from 0.0.0.0 for invalid RP 2.1.1.1
```

Vérifiez si le router2 est le RP pour le groupe 224.1.1.1 :

```
router2#show ip pim rp map
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 1.1.1.1 (?), v2v1
    Info source: 1.1.1.1 (?), elected via Auto-RP
    Uptime: 00:21:26, expires: 00:02:24
router2#
```

Le RP pour 224.1.1.1 est 1.1.1.1.

Vérifiez si c'est l'une des interfaces de router2 :

```
router2#show ip interface brief
Interface                IP-Address      OK? Method Status      Protocol
Ethernet0/0              1.1.1.2         YES NVRAM    up          up
Ethernet1/0              2.1.1.1         YES NVRAM    up          up
Ethernet2/0              unassigned      YES NVRAM    administratively down down
router2#
```

Puisque le router2 n'est pas un RP, il devrait ne pas avoir reçu ce paquet de connexion RP.

Vérifiez pourquoi le routeur en aval a envoyé 'connexion' au router2, alors qu'il ne devrait pas :

```
router3#show ip pim rp map
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 1.1.1.1 (?), v2v1
    Info source: 1.1.1.1 (?), elected via Auto-RP
    Uptime: 00:24:30, expires: 00:02:16
Group(s): 224.0.0.0/4, Static-Override
  RP: 2.1.1.1 (?)
router3#
```

Comme vous voyez, router3 a statiquement configuré les informations et des points RP à router2, qui est incorrect. Ceci explique pourquoi router3 envoie RP-se joint à router2.

## Correctifs possibles

Faites du router2 le RP pour le groupe 224.1.1.1 ou modifiez la configuration sur router3 de sorte qu'il se rapporte à l'adresse correcte RP.

```
router3#show run | i rp
ip pim rp-address 2.1.1.1 override
router3#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
router3(config)#no ip pim rp-address 2.1.1.1 override
router3(config)#exit
router3#
```

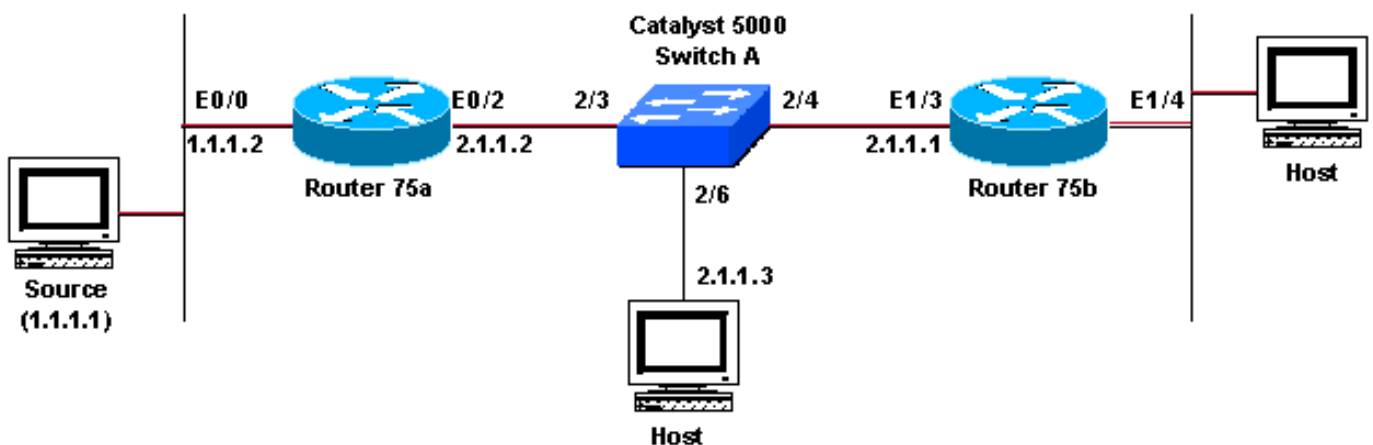
Après que la configuration sur router3 soit corrigée, le router3 se rapporte à l'adresse correcte RP, et le message d'erreur cesse d'apparaître.

```
router3#show ip pim rp map
PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4
  RP 1.1.1.1 (?), v2v1
    Info source: 1.1.1.1 (?), elected via Auto-RP
      Uptime: 00:30:45, expires: 00:02:02
router3#
```

## Le CGMP ne prévient pas la pléthore de paquets de multidiffusion

Cette section explique comment l'engorgement par paquets de multidiffusion non désiré peut se produire quand le protocole de gestion de groupe Cisco (CGMP) n'est pas activé sur tous les routeurs d'un sous-réseau particulier, et comment ce problème peut être évité. Ce schéma de réseau est utilisé comme exemple.



### diagnostic du problème

Dans la figure, la table CAM statique dans le commutateur A de Catalyst 5000 n'affiche pas les ports appropriés l'un des qui sont remplis. Le routeur configuré Cgmp n'envoie pas des paquets CGMP.

Le CGMP est correctement configuré avec la commande **régler cgmp à actif** sur le commutateur A et la commande **ip cgmp** sur l'interface E0/2 du routeur 75a. Cependant, aucun groupe de multidiffusion n'est visible sur le commutateur A quand la commande **show multicast group** est

émise :

```
Console> (enable) show multicast group
CGMP enabled
IGMP disabled
IGMP fastleave disabled
GMRP disabled
```

```
VLAN  Dest MAC/Route Des  [CoS]  Destination Ports or VCs / [Protocol Type]
-----
```

Total Number of Entries = 0

Le résultat de cette commande devrait montrer chaque port qui a un routeur configuré Cgmp (port 2/3) et chaque port qui a un récepteur intéressé (port 2/6). Puisque 0 est listé, vous pouvez supposer que tous les ports sont inondés inutilement avec le trafic de multidiffusion, qu'ils le veuillent ou pas.

## Observations

Vérifiez s'il y a des voisins multicast indépendants du protocole (PIM) sur le routeur 75a :

```
ip22-75a#show ip pim neighbor
PIM Neighbor Table
Neighbor Address  Interface          Uptime    Expires    Ver  Mode
2.1.1.1           Ethernet0/2        00:07:41  00:01:34  v2
```

La sortie prouve que le routeur 75a peut voir le routeur 75b en tant que voisin valide PIM par le commutateur A.

Vérifiez maintenant si vous recevez les informations correctes de route multicast (mroute) sur les Routeurs :

```
ip22-75a#show ip mroute 224.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 00:14:55/00:02:59, RP 0.0.0.0, flags: DJC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/2, Forward/Sparse-Dense, 00:14:55/00:00:00

(1.1.1.1, 224.1.1.1), 00:14:56/00:02:59, flags: CTA
  Incoming interface: Ethernet0/1, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0/2, Forward/Sparse-Dense, 00:14:56/00:00:00
```

La sortie affiche au routeur 75a en avant les paquets de multidiffusion E0/2 vers le commutateur A. Cette sortie affiche que le routeur 75b obtient les paquets de multidiffusion et en avant eux correctement :

```
ip22-75b#show ip mroute 224.1.1.1
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
```

A - Advertised via MSDP

Timers: Uptime/Expires

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

(\* , 224.1.1.1), 00:17:57/00:02:59, RP 0.0.0.0, flags: DJC

Incoming interface: Null, RPF nbr 0.0.0.0

Outgoing interface list:

Ethernet1/3, Forward/Sparse-Dense, 00:17:57/00:00:00

Ethernet1/4, Forward/Sparse-Dense, 00:17:57/00:00:00

(1.1.1.1, 224.1.1.1), 00:00:35/00:02:59, flags: CTA

Incoming interface: Ethernet1/3, RPF nbr 2.1.1.2

Outgoing interface list:

Ethernet1/4, Forward/Sparse-Dense, 00:00:35/00:00:00

Du point de vue du commutateur A, vous voyez qu'il voit le routeur 75a hors fonction du port 2/3.

```
Console> (enable) show multicast router
```

```
CGMP enabled
```

```
IGMP disabled
```

```
IGMP fastleave disabled
```

```
Port      Vlan
-----  -
2/3      6
```

```
Total Number of Entries = 1
```

Jusqu'ici, tout semble être en ordre. Sélectionnez quelques commandes de **débogage** sur le routeur 75a afin de voir ce que vous pouvez découvrir :

```
ip22-75a#debug ip cgmp
```

```
CGMP debugging is on
```

```
*Feb  8 12:45:22.206: CGMP: Sending self Join on Ethernet0/2
```

```
*Feb  8 12:45:22.206:      GDA 0000.0000.0000, USA 00d0.ff2f.a002
```

```
*Feb  8 12:46:22.234: CGMP: Sending self Join on Ethernet0/2
```

```
*Feb  8 12:46:22.234:      GDA 0000.0000.0000, USA 00d0.ff2f.a002
```

```
*Feb  8 12:47:22.262: CGMP: Sending self Join on Ethernet0/2
```

```
*Feb  8 12:47:22.262:      GDA 0000.0000.0000, USA 00d0.ff2f.a002
```

Dans la sortie, 0000.0000.0000 est les tout-groupes adresse et est utilisé quand les Routeurs envoient des messages de CGMP Join/congé ainsi les Commutateurs peuvent remplir ports de routeur. GDA représente l'adresse de destination de groupe dans le format de niveau de contrôle d'accès au support (MAC) et USA est l'acronyme de Unicast Source Address. C'est l'adresse de l'hôte qui a lancé le rapport IGMP pour lequel ce message de CGMP est généré. Dans ce cas, c'est l'adresse MAC pour l'interface E0/2 du routeur 75a. L'adresse MAC pour E0/2 du routage du routeur 75a peut être consultée avec la **commande show interface command** comme ici :

```
ip22-75a#show interface e0/2
```

```
Ethernet0/2 is up, line protocol is up
```

```
Hardware is cxBus Ethernet, address is 00d0.ff2f.a002 (bia 00d0.ff2f.a002)
```

En outre, vous pourriez périodiquement voir ceci quand la commande d'**igmp d'IP de débogage** est activée :

```
*Feb  8 12:51:41.854: IGMP: Received v2 Report from 2.1.1.3 (Ethernet0/2) for 224.1.1.1
```

Cependant, vous ne pouvez pas voir ultérieurement un paquet CGMP correspondant du routeur 75a. Ceci signifie que le routeur 75a reçoit des rapports IGMP, mais ne génère pas les paquets CGMP nécessaires afin d'aider le commutateur A pour connaître quels ports à bloquer. C'est ce qui serait prévu du routeur 75a si c'est le requérant d'IGMP. Cette sortie du routeur 75a nous indique pourquoi le comportement prévu ne se produit pas :

```
ip22-75a#show ip igmp interface e0/2
```

```
Ethernet0/2 is up, line protocol is up
  Internet address is 2.1.1.2/24
  IGMP is enabled on interface
  Current IGMP version is 2
  CGMP is enabled
  IGMP query interval is 60 seconds
  IGMP querier timeout is 120 seconds
  IGMP max query response time is 10 seconds
  Last member query response interval is 1000 ms
  Inbound IGMP access group is not set
  IGMP activity: 3 joins, 1 leaves
  Multicast routing is enabled on interface
  Multicast TTL threshold is 0
  Multicast designated router (DR) is 2.1.1.2 (this system)
  IGMP querying router is 2.1.1.1
  No multicast groups joined
```

Si vous avez deux routeurs sur le même sous-réseau, et si vous configurez chacun des deux pour le CGMP, seulement l'un des deux enverra des paquets CGMP. Le routeur qui envoie les paquets CGMP est le routeur auteur de la requête IGMP. Ceci signifie le routeur avec la plus basse adresse IP de monodiffusion des routeurs prenant en charge l'IGMP.

Dans ce cas, les routeurs 75a et 75b prennent en charge IGMP (le routeur 75b est devenu le routeur auteur de la requête IGMP), mais seulement le routeur 75a prend en charge le CGMP. Puisque le routeur 75a n'est pas le routeur auteur de la requête IGMP, aucun paquet CGMP n'est envoyé.

## Correctifs possibles

Afin de résoudre le problème, vous devez configurer le CGMP sur le routeur auteur de la requête IGMP. Dans ce cas, routeur 75b. D'abord, activez les commandes de débogage sur le routeur 75b :

```
ip22-75b#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ip22-75b(config)#int e 1/3
ip22-75b(config-if)#ip cgmp
ip22-75b(config-if)#end
ip22-75b#debug ip cgmp
ip22-75b#debug ip igmp
*Feb  8 12:58:42.422: IGMP: Received v2 Report from 2.1.1.3 (Ethernet1/3) for 224.1.1.1
*Feb  8 12:58:42.422: CGMP: Received IGMP Report on Ethernet1/3
*Feb  8 12:58:42.422:      from 2.1.1.3 for 224.1.1.1
*Feb  8 12:58:42.422: CGMP: Sending Join on Ethernet1/3
*Feb  8 12:58:42.422:      GDA 0100.5e01.0101, USA 0030.b655.a859
```

Le routeur 75b reçoit un rapport IGMP de 2.1.1.3 pour le groupe 224.1.1.1. Il envoie ultérieurement une demande de connexion CGMP au commutateur A pour l'équivalent MAC de 224.1.1.1 avec l'adresse MAC (USA) de l'hôte intéressé 2.1.1.3. Le commutateur A sait quel port qui est l'hôte est allumé et le marque comme ouvert et bloque les autres.

Maintenant tout devrait fonctionner correctement au commutateur A :

```
Console> (enable) show multicast group
CGMP enabled
IGMP disabled
IGMP fastleave disabled
GMRP disabled
```

```
VLAN  Dest MAC/Route Des      [CoS]  Destination Ports or VCs / [Protocol Type]
```

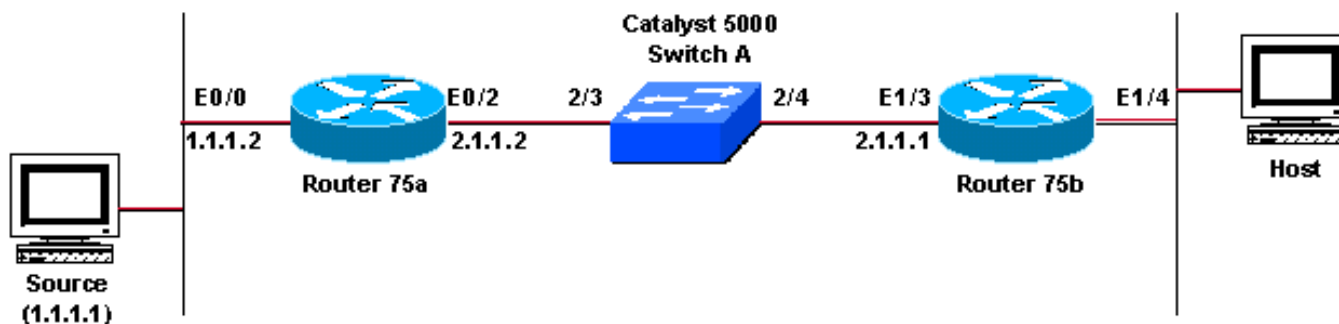
6	01-00-5e-00-01-28	2/3-4
6	01-00-5e-01-01-01	2/3-4, 2/6

Total Number of Entries = 2

C'est bien mieux. Les paquets de 224.1.1.1 (01-00-5e-01-01-01) sont seulement les ports envoyés 2/3, 2/4, et 2/6 sur le commutateur A, comme ils devraient. L'inondation des autres ports a arrêté. Le nombre total d'entrées est maintenant listé correctement comme 2. L'adresse MAC 01-00-5e-00-01-28 est mappée de l'adresse de multidiffusion 224.0.1.40 utilisée pour les connexions automatiques de CGMP.

## Le CGMP n'empêche pas l'engorgement par paquets de multidiffusion, dû à emplacement de source/récepteur

Cette section fournit une solution au problème courant d'un commutateur de Catalyst qui inonde le trafic à chaque port, au lieu de juste à l'hôte correct. Ce schéma de réseau est utilisé comme exemple.



### diagnostic du problème

Dans la figure, Routeurs 75a et 75b et le Catalyst 5000 (le commutateur A) sont correctement configurés pour la Multidiffusion et le Protocole CGMP (Cisco Group Management Protocol). L'hôte obtient le trafic de multidiffusion. Cependant, est ainsi chaque autre hôte hors fonction de commutateur A du commutateur A. inonde le trafic chaque port, ainsi il signifie que le CGMP ne fonctionne pas.

Le résultat de la commande **show multicast group** sur le commutateur A ressemble à :

```
Console> (enable) show multicast group
CGMP enabled
IGMP disabled
IGMP fastleave disabled
GMRP disabled
```

VLAN	Dest MAC/Route Des	[CoS]	Destination Ports or VCs / [Protocol Type]
6	01-00-5e-00-01-28		2/3-4

Total Number of Entries = 1

Vous pouvez voir de la sortie que le seul groupe est 224.0.1.40, qui est utilisé par le routeur quand il envoie le CGMP auto-se joint pour le groupe d'auto-RP. Comment se fait-il qu'il n'y ait aucun autre groupe ?

## Correctifs possibles

Afin de comprendre la solution, vous devez comprendre comment le CGMP se comporte dans certaines conditions. Un routeur Cgmp-activé envoie le CGMP se joint à un commutateur pour informer le commutateur des hôtes intéressés à un groupe de multidiffusion particulier. Le commutateur consulte les adresses MAC pour ces hôtes dans son tableau CAM, transfère alors des paquets de multidiffusion vers les ports avec des hôtes intéressés et bloque le transfert de tous les autres ports des paquets de multidiffusion.

Un routeur envoie une interface prenant en charge des connexions automatiques CGMP d'une interface prenant en charge le CGMP s'il reçoit des paquets de multidiffusion d'une source qui est sur la même interface que celle où le CGMP est pris en charge.

Par exemple, si la source était sur le même sous-réseau (VLAN), 2.1.1.0/24, que les routeurs 75a et 75b, CGMP fonctionnerait parfaitement. Le routeur, en consultant des paquets de la source, enverrait une demande de connexion automatique CGMP au commutateur, qui apprendrait dynamiquement sur quels ports le routeur fonctionne et qui bloquerait tous les autres ports des paquets de multidiffusion de transfert.

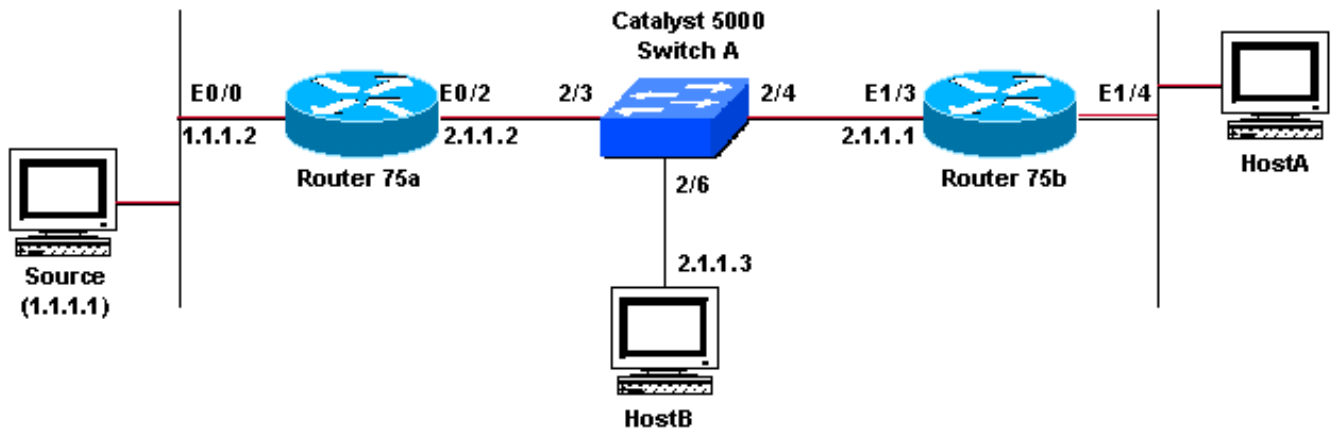
Un routeur envoie les demandes de connexion CGMP à une interface prenant en charge CGMP si elle reçoit des rapports IGMP d'un hôte sur la même interface que celle où le CGMP est pris en charge.

Un autre exemple est si vous aviez un hôte intéressé sur ce même VLAN. Dans ce cas, quand un routeur Cgmp-activé a reçu un état de Protocole IGMP (Internet Group Management Protocol) d'un hôte qui est intéressé par un groupe de multidiffusion particulier, le routeur enverrait un CGMP se joignent. La demande de connexion indiquerait l'adresse de contrôle d'accès au support (MAC) de l'hôte qui a voulu se connecter et du groupe auquel il a voulu se connecter. Le Catalyst 5000 vérifierait alors son tableau CAM pour l'adresse MAC de l'hôte et mettrait le port associé sur la liste de groupe de multidiffusion en bloquant tous les autres ports indifférents.

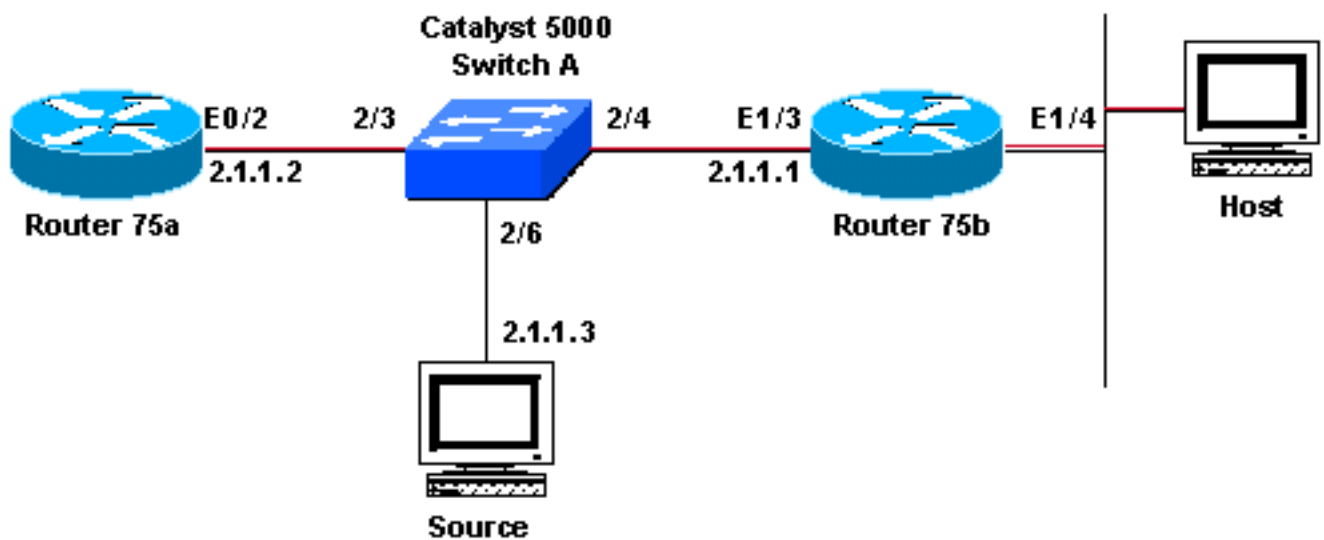
Ce n'est pas le cas quand la source et l'hôte intéressé sont sur un sous-réseau autre que le sous-réseau prenant en charge le CGMP (VLAN). Les paquets de multidiffusion, cela proviennent la source, ne déclenchent pas le routeur pour envoyer le CGMP auto-se joint au commutateur. Par conséquent, les paquets frappent le commutateur et inondent tout le VLAN. Ce scénario continue jusqu'à un hôte dans le VLAN, cela se dégage un port sur le commutateur, envoie un IGMP se joignent. Seulement avec la réception d'un rapport IGMP, le routeur enverra un paquet CGMP, et le commutateur ajoutera alors le port hôte approprié comme port transférant et tous les autres sont bloqués (hormis les ports du routeur).

Donc, pour que le CGMP fonctionne dans cette topologie de transit-type, vous pourriez ajouter un hôte au même VLAN que les routeurs 75a et 75b, comme dans ce diagramme de réseau.





Ou alors vous pouvez déplacer la source sur le même sous-réseau que les routeurs 75a et 75b, comme dans cet exemple.



Déplacez la source au même sous-réseau, puis vérifiez le résultat du commutateur A :

```

Console> (enable) show multicast router
CGMP enabled
IGMP disabled
IGMP fastleave disabled

```

Port	Vlan
2/3	6
2/4	6

```

Total Number of Entries = 2
'*' - Configured
Console> (enable)

```

```

Console> (enable) show multicast group
CGMP enabled
IGMP disabled
IGMP fastleave disabled
GMRP disabled

```

VLAN	Dest MAC/Route Des	[CoS]	Destination Ports or VCs / [Protocol Type]
6	01-00-5e-00-01-28		2/3-4

Total Number of Entries = 2

224.1.1.1 (01-00-5e-01-01-01) est maintenant seulement inondé aux ports du routeur 2/3 et 2/4 et non pas à chaque port du commutateur R.

## [Le CGMP n'empêche pas l'engorgement par paquets de multidiffusion pour certaines adresses de groupe](#)

Cette section explique pourquoi quelques adresses IP multicast causent l'inondation du trafic multidiffusion de tous les ports sur un LAN par le protocole de gestion de groupe Cisco. Quand vous utilisez l'adresse 225.0.0.1 de groupe de multidiffusion, le CGMP ne fonctionne pas. Il inonde le flux multidiffusion de tous les ports de commutation et gaspille la largeur de bande. Cependant, si vous modifiez l'adresse à 225.1.1.1, CGMP fonctionnera correctement. Quel est le problème avec utiliser 225.0.0.1, puisque ce n'est pas une adresse enregistrée pour un protocole de routage ?

### [Correctifs possibles](#)

D'abord, il est important de comprendre comment des adresses de multidiffusion de niveau 3 sont mappées pour les adresses de multidiffusion de niveau 2. Toutes les trames de Protocole IP Multicast utilisent les adresses de couche MAC d'IEEE qui commencent par le préfixe 24-bit de 0x0100.5e. Quand vous tracez la couche 3 pour poser 2 adresses, les 23 bits d'ordre réduit de l'adresse de multidiffusion de la couche 3 sont tracés dans les 23 bits d'ordre réduit de l'adresse MAC d'IEEE.

Un autre important fait à comprendre est là sont 28 bits d'espace adresse d'adresse unique pour une adresse de Protocole IP Multicast (32 bits sans les 4 premiers bits qui contiennent le préfixe de 1110 classes D). Puisqu'il y a seulement 23 bits branchés à l'adresse MAC d'IEEE, 5 bits de superposition demeurent. Ceci signifie que plusieurs adresses de niveau 3 de multidiffusion peuvent mapper à la même adresse de multidiffusion de niveau 2.

Exemple :

```
224.0.0.1 = 1110 0000.0000 0000.0000 0000.0000 0001 in binary
low order 23 bits =    000 0000.0000 0000.0000 0001
hex equivalent    =    0   0   0   0   0   1
result of mapping = 0x0100.5e00.0001
```

```
224.128.0.1 = 1110 0000.1000 0000.0000 0000.0000 0001 in binary
low order 23 bits =    000 0000.0000 0000.0000 0001
hex equivalent    =    0   0   0   0   0   1
result of mapping = 0x0100.5e00.0001
```

Notez dans l'exemple 224.0.0.1 et carte de 224.128.0.1 à la même chose adresse de multidiffusion de la couche 2.

Maintenant que vous savez la couche 3 poser 2 adresses de multidiffusion sont tracées, poursuivez à la solution spécifique au ce problème.

Le commutateur A inonde des paquets de multidiffusion à 224.0.0.x parce que ces adresses sont des lien-gens du pays et vous voulez veiller des adresses locales à la liaison pour obtenir à tous les périphériques sur le réseau local (RÉSEAU LOCAL). Les commutateurs Catalyst inondent

également des paquets de multidiffusion à d'autres adresses de multidiffusion qui sont ambiguës de niveau avec 224.0.0.x (par exemple, 224.0.0.1 et 225.0.0.1 mappent les deux à 0100.5e00.0001). Le commutateur inonde ces adresses de liaison locales de paquets de multidiffusion que le CGMP soit activé ou non.

Par conséquent, l'application multidiffusion doit éviter l'utilisation des adresses de la classe D qui mappent à une adresse de multidiffusion de niveau 2 de 0100.5e00.00xx, où xx peut être 00 à FF en hexadécimales. Ceci se compose de ces adresses de la classe D :

```
224.0.0.x (x = 0 to 255)
225.0.0.x
.
239.0.0.x
```

```
224.128.0.x (x = 0 to 255)
225.128.0.x
.
239.128.0.x
```

## Des paquets de multidiffusion sont reçus en double

### Cause 1

Des paquets de multidiffusion sont reçus en double quand deux routeurs sont configurés en mode dense. En mode dense, le périphérique inonde périodiquement le flux. Après l'inondation, il taille des interfaces où le flux n'est pas souhaité. Les deux routeurs passent également par le processus d'assertion et décident qui est l'expéditeur. Chaque fois que les temporisateurs vont outre de ceci se produit, et jusqu'à ce processus sont complets, les deux Routeurs expédient le flot. A cause de cela, l'application reçoit les flux multidiffusion en double.

### Correctif possible 1

Ce problème peut être résolu si vous avez un des routeurs configurés pour le routage multicast et l'autre routeur pour être le RP en amont. Configurez-le afin de convertir le flux en mode intermédiaire avant que le flux n'entre dans le routeur. Ceci pour éviter que les paquets en double atteignent l'application. Ce n'est pas une la responsabilité de réseaux de garantir qu'il n'y a pas de paquet en double atteignant un hôte d'extrémité. C'est la responsabilité de l'application de prendre en charge des paquets en double et d'ignorer des données inutiles.

### Cause 2

Ce problème peut se produire dans des commutateurs de Cisco Catalyst 6500, qui sont configurés pour le mode de réplication multicast de sortie. Il peut être déclenché par suppression et réinsertion de toutes les cartes de ligne [OIR]. Après OIR, le port de matrice de la sortie [FPOE] peut misprogrammed, qui peut causer des paquets d'être dirigé vers le canal faux de sortie de matrice et d'être envoyé aux linecards faux. Le résultat est une boucle de retour à la structure et une réplication multiple quant ils sortent sur la carte correcte.

```
C6509#show mls ip multicast capability
Current mode of replication is Egress
Auto replication mode detection is ON
```

```
Slot          Multicast replication capability
```

1	Egress
2	Egress
3	Egress
4	Egress
5	Egress
6	Egress
7	Egress

## Correctif possible 2

Comme solution, modifiez le mode de réplication en entrée. Pendant une modification de réplication de sortie au mode réplication en entrée, des interruptions du trafic peuvent se produire parce que les raccourcis sont purgés et réinstallés.

```
mls ip multicast replication-mode ingress
```

Mettez à niveau le logiciel Cisco IOS à une version non affectée par le Cisco bug ID [CSCeg28814](#) (clients [enregistrés](#) seulement) afin de résoudre le problème une fois pour toutes.

## Cause 3

Cette question peut également se produire si la configuration de l'évolution de côté de réception (RSS), sur les fin-hôtes ou les serveurs, est désactivée.

## Correctif possible 3

Le paramètre RSS facilite une transmission plus rapide des données à travers plusieurs processeurs. Activez le paramètre RSS sur l'hôte final ou le serveur. Référez-vous au [réseau extensible](#) d'article de Microsoft [avec le](#) pour en savoir plus [RSS](#).

## Pourquoi est-ce que des paquets de multidiffusion sont déposés ?

### Cause 1

Il est possible que vous voyez les inondations excessives, et que le paquet en entrée soit déposé sur les interfaces pendant le trafic multidiffusion. Vous pouvez vérifier les inondations avec la **commande show interface**.

```
Switch#show interface gi 1/0
```

```
!--- Output suppressed MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec, reliability 255/255, txload 1/255, rxload 1/255 Encapsulation ARPA, loopback not set Keepalive set (10 sec) Full-duplex, 1000Mb/s, media type is SX input flow-control is off, output flow-control is on Clock mode is auto ARP type: ARPA, ARP Timeout 04:00:00 Last input 00:00:00, output 00:00:00, output hang never Last clearing of "show interface" counters never Input queue: 2/75/0/13370328 (size/max/drops/flushes); Total output drops: 0 Queueing strategy: fifo Output queue: 0/40 (size/max) 30 second input rate 195000 bits/sec, 85 packets/sec 30 second output rate 1000 bits/sec, 1 packets/sec L2 Switched: ucast: 53056 pkt, 4728434 bytes - mcast: 235759386 pkt, 66914376970 bytes L3 in Switched: ucast: 8714263 pkt, 1815426423 bytes - mcast: 1081138213 pkt, 438000092206 bytes mcast L3 out Switched: ucast: 4939256 pkt, 790351689 bytes mcast: 0 pkt, 0 bytes 1326976857 packets input, 506833655045 bytes, 0 no buffer Received 1318209538 broadcasts, 0 runts, 0 giants, 1 throttles 0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored 0 input packets with dribble condition detected 31643944 packets output, 3124494549 bytes, 0 underruns 0 output errors, 0 collisions, 1 interface resets 0 babbles, 0 late collision, 0 deferred 0 lost carrier, 0 no carrier 0 output buffer failures, 0 output buffers swapped out
```

## [Correctif possible 1](#)

Vous pouvez définir la valeur SPT comme mesure infinie pour l'interface où les inondations excessives se produisent.

Configurez ceci :

```
Switch(config-if)#ip pim spt-threshold infinity
```

## [Cause 2](#)

Quand vous utilisez la commande de **ip igmp join-group <group-name>** sur une des interfaces, elle commute le processus. Si les paquets de multidiffusion sont changés de processus sur une des interfaces, ils consomment plus de mémoire de processeur parce que tous les paquets de ce groupe seront commutés. Vous pouvez exécuter la commande **montrer buffers input-interface** et vérifier la taille anormale.

```
Switch#show buffers input-interface gi 1/0
```

Header	DataArea	Pool	Rcnt	Size	Link	Enc	Flags	Input	Output
437C6EAC	8096AE4	Middl	1	434	7	1	280	Gi1/1	None
437C74B4	8097864	Middl	1	298	7	1	280	Gi1/1	None
437C98E4	809C964	Middl	1	434	7	1	280	Gi1/1	None
437CAAFC	809F1E4	Middl	1	349	7	1	280	Gi1/1	None
437CAE00	809F8A4	Middl	1	519	7	1	280	Gi1/1	None

*!--- Output suppressed*

## [Correctif possible 2](#)

Vous pouvez utiliser la command **ip igmp static-group <group-name>** au lieu de la commande **ip igmp join-group <group-name>**.

Remarque: En raison des anciens numéros, il est possible que vous constatiez une utilisation de mémoire de processeur accrue d'environ 90 pour cent. Le processeur descend à la normale quand vous les résolvez le problème avec ces correctifs possibles.

## [Informations connexes](#)

- [Outils de dépannage de multidiffusion de base](#)
- [Guide de configuration pour le démarrage rapide de la multidiffusion](#)
- [Assistance pour la technologie multicast IP](#)
- [Assistance pour les protocoles de routage IP](#)
- [Soutien technique et documentation Cisco Systems](#)