

《IP组播故障排除指南》

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[由于 RPF 故障，路由器未向主机转发多播数据包](#)

[诊断问题](#)

[可能的修正](#)

[由于源端 TTL 设置，路由器未向主机转发多播数据包](#)

[诊断问题](#)

[可能的修正](#)

[由于路由器的 TTL 阈值，路由器未转发多播数据包](#)

[诊断问题](#)

[可能的修正](#)

[多个等价路径导致不需要的RPFBehavior](#)

[诊断问题](#)

[可能的修正](#)

[为什么没有在所有可用的等价路径之间进行 IP 多播负载均衡？](#)

[可能的修正](#)

[当您在路由器上收到IP组播“INVALID RP JOIN”错误消息时](#)

[诊断问题 - 第 1 部分](#)

[可能的修正](#)

[诊断问题 - 第 2 部分](#)

[可能的修正](#)

[收到重复的多播数据包流](#)

[原因 1](#)

[可能的修复方法 1](#)

[原因 2](#)

[可能的修复方法 2](#)

[原因 3](#)

[可能的修复方法 3](#)

[多播数据包为什么被丢弃？](#)

[原因 1](#)

[可能的修复方法 1](#)

[原因 2](#)

[可能的修复方法 2](#)

[相关信息](#)

简介

本文介绍 IP 多播的常见问题和解决方案。

先决条件

要求

本文档没有任何特定的要求。

使用的组件

本文档不限于特定的软件和硬件版本。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

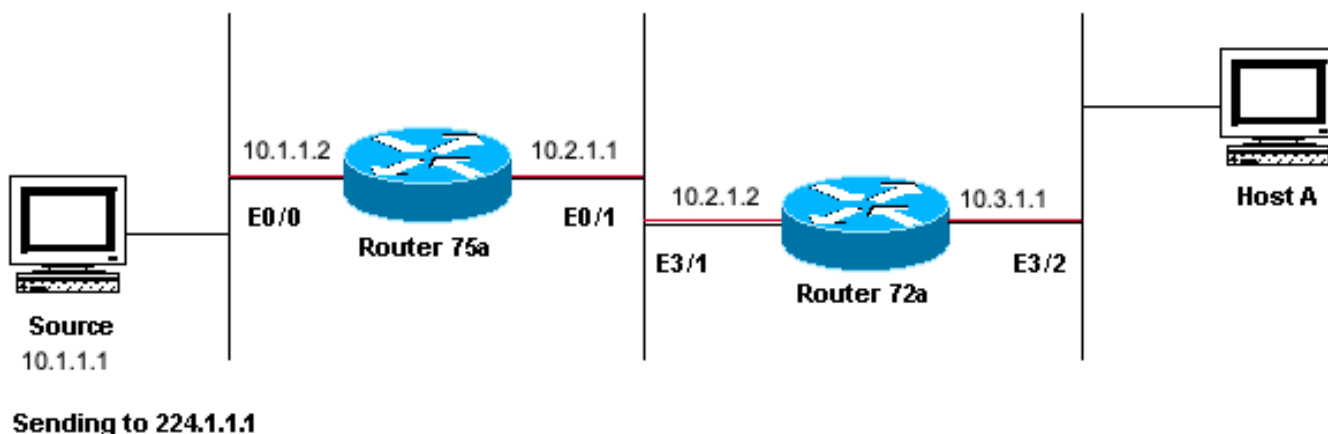
背景信息

当您排除组播路由故障时，主要关注点是源地址。组播具有“逆向转发” (RPF) 检查这一功能。当组播数据包到达接口时，RPF 进程会执行检查以确保此传入接口是单播路由所使用的传出接口，以便访问组播数据包的源。此 RPF 检查进程将防止出现环路。组播路由不会转发数据包，除非数据包的源通过了 RPF 检查。数据包通过此 RPF 检查后，多播路由将仅根据目标地址转发数据包。

与单播路由一样，组播路由也具有多种可用的协议，例如协议无关组播密集模式 (PIM-DM)、PIM 稀疏模式 (PIM-SM)、距离矢量组播路由协议 (DVMRP)、组播边界网关协议 (MBGP) 和组播源发现协议 (MSDP)。本文档中的各案例研究将引导您完成针对各种问题的故障排除过程。您可以查看使用哪些命令来快速查明问题并学习如何解决它。这里列出的案例研究适用于各种协议，除非特别说明。

由于 RPF 故障，路由器未向主机转发多播数据包

本节介绍如何解决常见的 IP 组播 RPF 故障问题。我们以下面的网络图为例。



在此图中，组播数据包从IP地址为10.1.1.1的服务器进入路由器75a的E0/0并发送到组224.1.1.1。这称为 (S,G) 或 (10.1.1.1, 224.1.1.1)。

诊断问题

直接连接到路由器 75a 的主机将接收该多播馈送，但直接连接到路由器 72a 的主机不会进行接收。首先，输入[show ip mroute](#)命令以检查路由器75a上的活动。该命令将检查组地址 224.1.1.1 的多播路由 (mroute)：

```
<#root>
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

(10.1.1.1, 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
```

由于路由器在 PIM 密集模式 (D 标志表示路由器处于密集模式) 下运行，因此请忽略 *,G 条目并关注 S,G 条目。该条目表明多播数据包来自地址为 10.1.1.1 的服务器并发送到多播组 224.1.1.1。数据包进入以太网接口 0/0，并从以太网接口 0/1 转发出去。这是一个完美的方案。

输入 show ip pim neighbor 命令，以查看路由器72a是否将上游路由器(75a)显示为PIM邻居：

```
<#root>
ip22-72a#
show ip pim neighbor

PIM Neighbor Table
Neighbor Address  Interface          Uptime    Expires    Ver  Mode
10.2.1.1         Ethernet3/1        2d00h     00:01:15  v2
```

从show ip pim neighbor命令的输出看，PIM邻居看起来一切正常。

输入 show ip mroute 命令，以查看路由器72a是否具有良好的多播路由：

```
<#root>
```

```
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
```

```
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:
```

通过show ip mroute 224.1.1.1 命令可以发现，传入接口为Ethernet2/0而不是预期的Ethernet3/1。

输入show ip mroute 224.1.1.1 count命令以查看该组是否有任何多播流量抵达路由器72a以及接下来发生的情况：

```
<#root>
```

```
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 F
```

```
Other: 471/471/0
```

```
ip22-72a#
```

从Other计数中可以看到流量由于RPF故障而被丢弃：由于RPF故障而丢弃的总数为471个- 471...

输入show ip rpf <source>命令以查看是否有RPF错误：

```
<#root>
```

```
ip22-72a#
```

```
show ip rpf 10.1.1.1
```

```
RPF information for ? (10.1.1.1) RPF interface: Ethernet2/0 RPF neighbor: ? (0.0.0.0) RPF route/mask:
```

Cisco IOS® 以这种方式计算 RPF 接口。可能的 RPF 信息来源包括单播路由表、MBGP 路由表、DVMRP 路由表和静态 Mroute 表。

在计算 RPF 接口时，主要使用管理距离来确定 RPF 计算过程所依据的信息源。具体规则如下：

•

系统会搜索所有以前的RPF数据源，以查找源IP地址的匹配项。使用共享树时，将使用 RP 地址而不是源地址。

-

如果找到多个匹配路由，则使用管理距离最小的路由。

-

如果管理距离相等，则使用以下优先顺序：

- 1.

静态多播路由

- 2.

DVMRP 路由

- 3.

MBGP 路由

- 4.

单播路由

-

如果某个路由在同一个路由表中有多个条目，则使用最长的匹配路由。

show ip mroute 224.1.1.1

命令输出表明RPF接口为E2/0，但传入接口必须是E3/1。

输入**show ip mroute 224.1.1.1** 命令以查看RPF接口为什么与预期接口不同。

<#root>

ip22-72a#

```
show ip route 10.1.1.1
```

```
Routing entry for 10.1.1.1/32 Known via "static", distance 1, metric 0 (connected) Routing Descriptor
```

从此 `show ip route 10.1.1.1` 命令的输出中可以看到有一个静态 /32 路由，它导致选择了错误的接口作为 RPF 接口。

进一步输入一些[调试命令](#)：


```
<#root>
```

```
ip22-72a#
```

```
debug ip mpacket 224.1.1.1
```

```
*Jan 14 09:45:32.972: IP: s=10.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
*Jan 14 09:45:33.020: IP: s=10.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
*Jan 14 09:45:33.072: IP: s=10.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
*Jan 14 09:45:33.120: IP: s=10.1.1.1 (Ethernet3/1)
d=224.1.1.1 len 60, not RPF interface
```

数据包在 E3/1 上传入，这是正确的。但是，这些数据包将被丢弃，因为这不是单播路由表用于执行 RPF 检查的接口。

 **注意：**调试数据包非常危险。数据包调试操作将触发组播数据包的进程交换，这会占用大量 CPU 资源。此外，数据包调试还会产生大量输出，由于向控制台端口输出过慢，从而可能导致将路由器完全挂起。要特别注意，必须在调试数据包前禁止将日志记录输出到控制台，并允许将日志记录转到内存缓冲区。为此，需配置 `no logging console` 和 `logging buffered` debugging。可以使用 `show logging` 命令查看调试结果。

可能的修正

您可以更改单播路由表以满足此要求，也可以添加静态 mroute 以强制组播从特定接口进行 RPF，而不管单播路由表状态如何。添加静态多播路由：

```
<#root>
```

```
ip22-72a(config)#
```

```
ip mroute 10.1.1.1 255.255.255.255 10.2.1.1
```

此静态mroute表明，要到达RPF的地址10.1.1.1，请使用10.2.1.1作为出接口E3/1的下一跳。

```
<#root>
```

```
ip22-72a#
```

```
show ip rpf 10.1.1.1
```

```
RPF information for ? (10.1.1.1)
RPF interface: Ethernet3/1
RPF neighbor: ? (10.2.1.1)
RPF route/mask: 10.1.1.1/32
RPF type: static mroute
RPF recursion count: 0
Doing distance-preferred lookups across tables
```

show ip mroute 和debug ip mpacket 的输出一切正常，show ip mroute count中的发送数据包数增加了，并且HostA收到数据包。

```
<#root>
```

```
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-b

```
show ip mroute 224.1.1.1 count
```

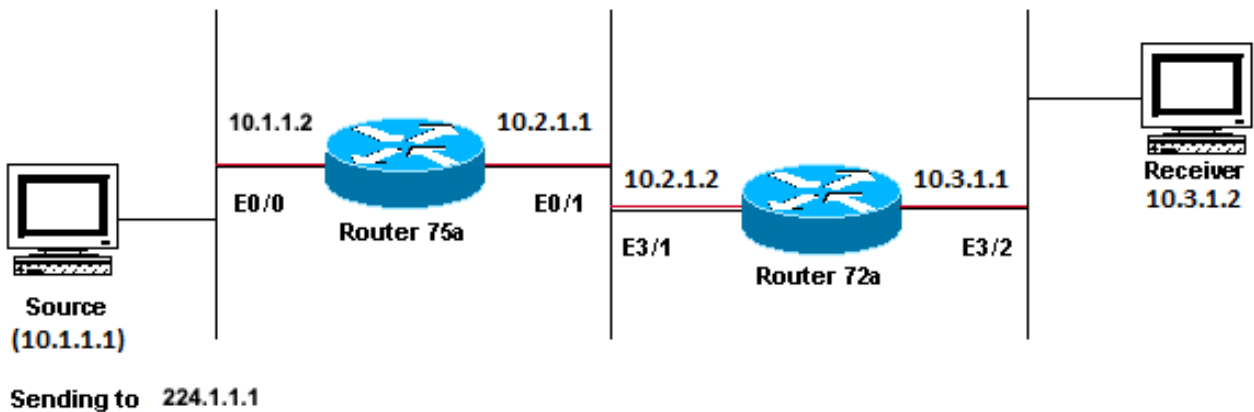
IP Multicast Statistics 3 routes using 2378 bytes of memory 2 groups, 0.50 average sources per group F

```
debug ip mpacket 224.1.1.1
```

```
*Jan 14 10:18:29.951: IP: s=10.1.1.1 (Ethernet3/1) d=224.1.1.1 (Ethernet3/2) len 60, mforward *Jan 14
```

由于源端 TTL 设置，路由器未向主机转发多播数据包

本节介绍如何解决因生存时间 (TTL) 值递减为零而无法转发 IP 组播数据包这一常见问题。由于多播应用的情况较多，因此这个问题很常见。多播应用主要设计用于 LAN，因此会将 TTL 设置为一个较低的值，甚至为 1。请以下的网络图为例。



主要针对LAN使用设计的组播应用示例

诊断问题

在上图中，路由器A不会将来自源的数据包转发到直接连接的接收方路由器B。查看路由器A的 `show ip mroute` 命令的输出以检查多播路由状态：

```
<#root>
```

```
ROUTERA#
```



```
show ip mroute
```

IP Multicast Routing Table Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned R - RP-b

可以忽略输出中的 224.0.1.40，因为所有路由器都加入此 Auto-RP 发现组。但是这里没有为 224.1.1.1 列出源。除了“*，224.1.1.1”以外，您还看不到“10.1.1.1，224.1.1.1”。

输入 `show ip rpf` 命令以查看这是否是 RPF 问题：

```
<#root>
```

```
ROUTERA#
```

```
show ip rpf 10.1.1.1
```

```
RPF information for ? (10.1.1.1)
RPF interface: Ethernet0/0
RPF neighbor: ? (0.0.0.0) - directly connected
RPF route/mask: 10.1.1.1/24
RPF type: unicast (connected)
RPF recursion count: 0
Doing distance-preferred lookups across tables
```

从输出中，它不是RPF问题。RPF检查正确指向E0/0以访问源IP地址。

使用`show ip pim interface`命令检查接口是否配置了PIM：

```
<#root>
```

```
ROUTERA#
```

```
show ip pim interface
```

| Address | Interface | Version/Mode | Nbr Count | Query Intvl | DR |
|----------|-------------|-----------------|--------------|----------------|----------|
| 10.1.1.2 | Ethernet0/0 | v2/Sparse-Dense | 0 | 30 | 10.1.1.2 |
| 10.2.1.1 | Ethernet0/1 | v2/Sparse-Dense | 2 | 30 | 10.2.1.2 |

输出一切正常，因此这不是问题所在。使用show ip mroute active命令检查路由器是否识别出任何活动流量：

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

输出内容表明，路由器无法识别任何活动流量。

```
<#root>
```

```
ROUTERA#
```

```
debug ip mpacket
```

```
IP multicast packets debugging is on
```

也许接收方不发送组224.1.1.1的任何Internet组管理协议(IGMP)报告（加入）。您可以使用show ip igmp group命令检查它：

```
<#root>
```

```
ROUTERB#
```

```
show ip igmp group
```

```
IGMP Connected Group Membership Group Address Interface Uptime Expires Last Reporter 224.0.1.40 Ethernet
```

224.1.1.1 已在 E1/2 加入，这是正确的。

PIM 密集模式是一个泛洪和修剪协议，因此没有加入，但是有嫁接。由于路由器 B 未从路由器 A 接收到泛洪，因此它并不知道向何处发送嫁接。

您可以使用嗅探器捕捉和 `show ip traffic` 命令进行检查以查看是否是 TTL 问题：

```
<#root>
```

```
ROUTERA#
```

```
show ip traffic
```

```
IP statistics: Rcvd: 248756 total, 1185 local destination 0 format errors, 0 checksum errors, 63744 ba
```

输出中显示了 63744 次不良跳数。每次键入此命令时，坏跳计数都将增加。这明确表示，发送方在 TTL 为 1 (路由器 A 将其递减为零) 的情况下发送数据包。这导致坏跳计数字段不断增加。

可能的修正

要解决此问题，您需要增大 TTL。这要在发送方的应用级别进行设置。有关详细信息，请参阅您的多播应用说明手册。

完成设置后，路由器 A 将如下所示：

```
<#root>
```

```
ROUTERA#
```

```
show ip mroute
```

```
IP Multicast Routing Table Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned R - RP-b
```

这是您希望的输出。

在路由器 B 上，您会看到：

```
<#root>
```

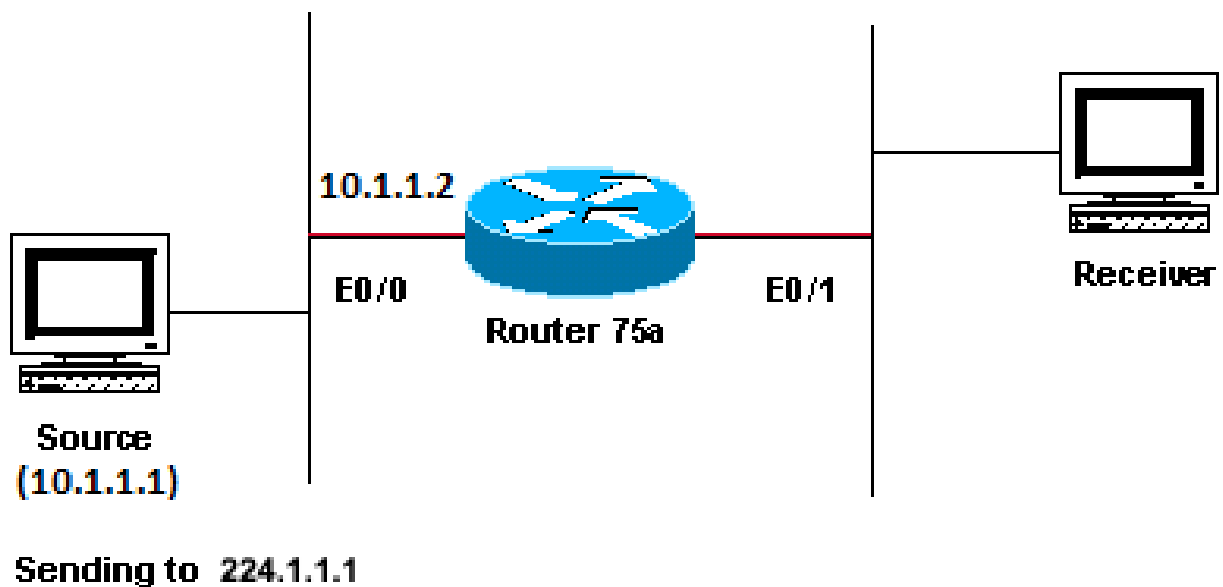
```
ROUTERB#
```

```
show ip mroute
```

IP Multicast Routing Table Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned R - RP-b

由于路由器的 TTL 阈值，路由器未转发多播数据包

本节介绍如何解决因 TTL 阈值设得过低而使 IP 组播流量无法到达接收方这一常见问题。我们以下面的网络图为例。



TTL 阈值太低，因此 IP 组播流量无法到达接收方

诊断问题

在上图中，接收方无法接收来自源的组播数据包。源设备和路由器 75a 之间可以有多个路由器。首先查看路由器 75a，因为它直接连接到接收方。

```
<#root>
```

```
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-b

输出内容显示，路由器 75a 从以太网接口 0/1 转发数据包。为了绝对确定路由器75a是否转发了数据包，请仅debug 为此源和多播组打开：

```
<#root>
```

```
ip22-75a#
```

```
configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z. ip22-75a(config)#

```
access-list 101 permit udp host 10.1.1.1 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=10.1.1.1.
```

debug 消息表明路由器75a并未转发多播数据包，因为已达TTL阈值。查看路由器配置，看看是否可以找到原因。下面的输出表明了问题所在：

```
interface Ethernet0/1 ip address 10.2.1.1 255.255.255.0 ip pim sparse-dense-mode ip multicast ttl-threshold 15
```

路由器的 TTL 阈值为 15，但这并不意味着任何大于 15 的 TTL 都不会予以发送。事实恰恰相反。应用是在 TTL 为 15 的情况下发送的。当它到达路由器 75a 时，多播数据包的 TTL 将小于 15。

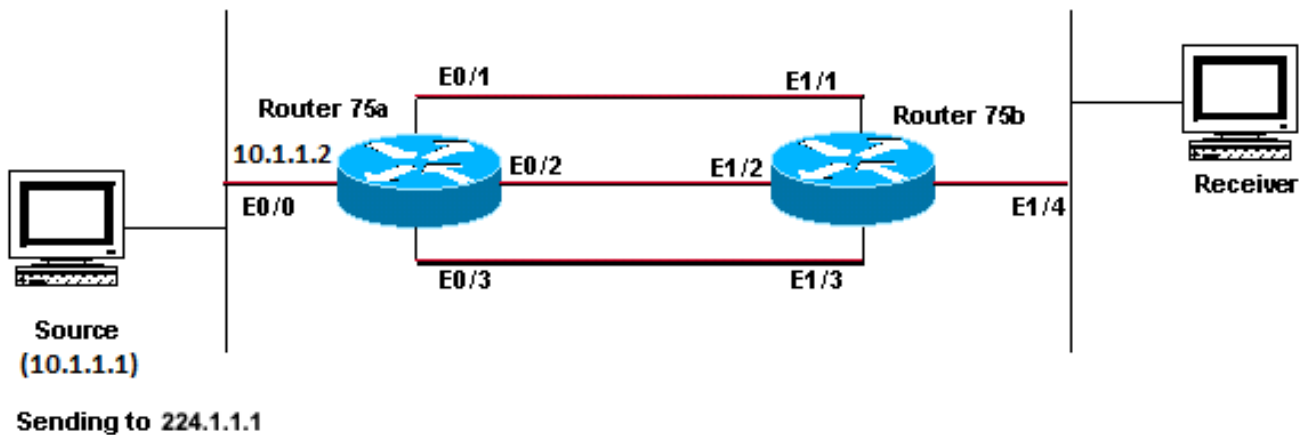
ip multicast ttl-threshold <value>命令意味着TTL低于指定阈值（本例中为 15）的任何数据包都不会进行转发。此命令通常用于提供边界，以防止内部组播流量从内联网流出。

可能的修正

请使用ip multicast ttl-threshold <value>命令的形式（这将恢复默认TTL阈值0）删除命令，或者降低TTL阈值以便能够传递流量。

多条等价路径导致意外 RPF 行为

本节介绍组播源的等价路径如何导致意外 RPF 行为。此外，还介绍如何配置 IP 组播以避免此行为。我们以下的网络图为例。



指向组播源的等价路径导致不需要的RPF行为

诊断问题

在图中，路由器75b具有返回源(10.1.1.1)的三个等价路径，并且它选择您不希望作为首选链路的链路作为RPF链路。

当面对指向源的多个等价路径时，IP 多播会选择其独立多播协议 (PIM) 邻居具有最高 IP 地址的接口作为传入接口，然后将修剪发送给其他链路上的 PIM 邻居。

可能的修正

要更改 IP 组播选为其传入接口的接口，可采取以下任一项操作：

-

只在希望多播流经的接口上配置 PIM，这意味着要牺牲多播冗余。

-

更改子网，以使最高 IP 地址位于您希望作为多播主链路的链路上。

-

创建一个指出首选组播接口的静态组播路由 (mroute)，这意味着您将失去组播冗余。

为举例说明，我们将创建一个静态多播路由。

在安装静态 mroute 前，此输出中显示，路由表中有源地址 10.1.1.1 的三条等价路由。RPF 信息表明 RPF 接口为 E1/3：

```
<#root>
```

```
ip22-75b#
```

```
show ip route 10.1.1.1
```

```
Routing entry for 10.1.1.1/24 Known via "ospf 1", distance 110, metric 20, type intra area Redistribut
```

配置静态 mroute 后，此输出中显示，RPF 接口已更改为 E1/1：

```
<#root>
```

```
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 10.2.1.1
```

```
ip22-75b(config)#end ip22-75b#
```

```
show ip rpf 10.1.1.1
```

RPF information for ? (10.1.1.1) RPF interface: Ethernet1/1 RPF neighbor: ? (10.2.1.1) RPF route/mask:

```
show ip route 10.1.1.1
```


Routing entry for 10.1.1.1/24 Known via "ospf 1", distance 110, metric 20, type intra area Redistribut

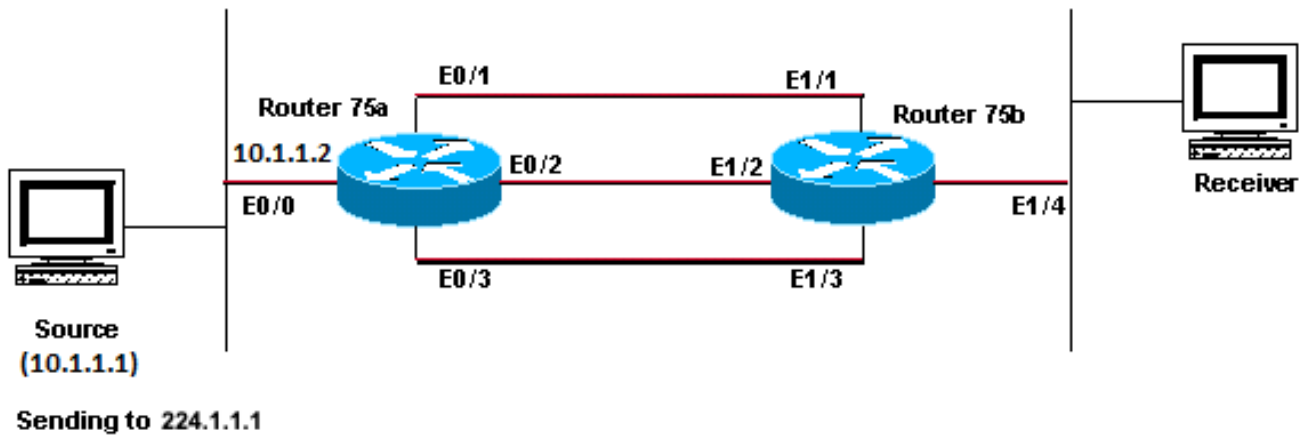
```
show ip mroute 224.1.1.1
```

IP Multicast Routing Table Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned R - RP-b

为什么没有在所有可用的等价路径之间进行 IP 多播负载均衡？

本节提供一个解决方案，解决有关如何配置 IP 组播以在所有可用等价路径上实现负载均衡这一常见问题。我们以下面的网络图为例。

 **注意：**在通过隧道在等价路径上对IP组播流量进行负载分流之前，请配置CEF每数据包负载均衡，否则GRE数据包无法按数据包进行负载均衡。有关在组播环境中执行负载分流的其他方法，请参阅[通过 ECMP 拆分 IP 组播流量](#)。



配置IP组播以跨所有可用的等价路径实现负载均衡

在图中，路由器75b有三个返回源(10.1.1.1)的等价路径。您希望在全部三条链路上均衡多播流量。

可能的修正

如[多条等价路径导致意外RPF行为一节中所述](#)，协议无关组播(PIM)仅会选择一个接口用于进行RPF检查，并删除其他接口。这意味着不会进行负载均衡。要进行负载均衡，需要从冗余链路中删除PIM并在路由器75a与路由器75b之间创建一条隧道。然后您可以在链路级别进行负载均衡，并且IP将通过隧道运行。

下面是隧道的配置。

| 路由器 75a |
|--|
| <pre> interface Tunnel0 ip address 10.6.1.1 255.255.255.0 ip pim sparse-dense-mode tunnel source Ethernet0/0 tunnel destination 10.5.1.1 ! interface Ethernet0/0 ip address 10.1.1.2 255.255.255.0 ip pim sparse-dense-mode ! interface Ethernet0/1 ip address 10.2.1.1 255.255.255.0 ! interface Ethernet0/2 ip address 10.3.1.1 255.255.255.0 ! </pre> |

```
interface Ethernet0/3
ip address 10.4.1.1 255.255.255.0
```

路由器 75b

```
interface Tunnel0
ip address 10.6.1.2 255.255.255.0
ip pim sparse-dense-mode
tunnel source Ethernet1/4
tunnel destination 10.1.1.2
!
interface Ethernet1/1
ip address 10.2.1.2 255.255.255.0
!
interface Ethernet1/2
ip address 10.3.1.2 255.255.255.0
!
interface Ethernet1/3
ip address 10.4.1.2 255.255.255.0
!
interface Ethernet1/4
ip address 10.5.1.1 255.255.255.0
ip pim sparse-dense-mode
!
ip mroute 0.0.0.0 0.0.0.0 Tunnel0
```

配置隧道后，输入show ip mroute命令查看组的多播路由(mroute)：

```
<#root>
```

```
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-b

```
show ip mroute 224.1.1.1
```

IP Multicast Routing Table Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned R - RP-b

为了检查组播数据是否在三条链路上均等地平衡负载，请查看路由器 75a 的接口数据。

传入接口为 9000 位/秒：

```
<#root>
```

```
ip22-75a#
```

```
show interface e0/0
```

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

三个传出接口每个均显示 3000 位/秒：

```
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 pac
```

当您在路由器上收到IP组播“INVALID_RP_JOIN”错误消息时

此部分为常见的 IP 多播“INVALID_RP_JOIN”错误消息提供了一个解决方案。

诊断问题 - 第 1 部分

在交汇点 (RP) 上收到此错误消息：

```
00:55:15: %PIM-6-INVALID_RP_JOIN: Received (*, 224.1.1.1) Join from 10.1.6.2 for invalid RP 10.1.5.4 00:56:15: %PIM-6-INVALID_RP_JOIN: Rec
```

Cisco IOS Software System Error Messages (Cisco IOS软件系统错误消息) 文档解释了此错误的原因：下游PIM路由器为共享树发送了加入消息，此路由器不想接受。这一行为表明此路由器只允许下游路由器加入特定 RP。

怀疑它是过滤。您需要查看一下路由器的配置：

```
interface Ethernet0/0 ip address 10.1.5.4 255.255.255.0 ip pim sparse-dense-mode ! ip pim accept-rp 10.2.2.2 8 ip pim send-rp-announce Ethernet0/0 scope
```

配置中的 **accept-rp** 语句的用途是什么？在IP多播路由命令中，它指出“要配置路由器以接受指定RP和特定组列表的‘加入’或‘修剪’设置，请使用ip pim accept-rp全局配置命令。要删除这项检查，可使用此命令的 **no** 格式。”

如果删除 ip pim accept-rp命令，消息会消失。已找到了导致问题的命令，但是如果要在配置中保留该命令，该怎么办？您可以允许错误的RP地址。输入 **show ip pim rp mapping** 命令以查看正确的RP地址：

```
<#root>
```

```
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
```

根据输出，10.1.5.4是通过自动RP或其他方式获知的唯一RP。然而，此路由器是组 224.0.0.0/4 的唯一 RP。因此，配置中的 pim accept-rp语句是错误的。

可能的修正

解决方案是纠正ip pim accept-rp语句中的IP地址，如下所示：

将此语句：

```
<#root>
```

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

执行此更改：

```
<#root>
```

```
ip pim accept-rp 10.1.5.4 8
```

您还可以更改语句以接受 auto-rp 缓存中的内容，并确保访问列表（在本示例中为 8）允许必要的组播组范围。示例如下：

```
<#root>
```

```
ip pim accept-rp auto-rp access-list 8 permit 224.0.0.0 0.255.255.255
```

诊断问题 - 第 2 部分

此错误消息出现在 router2 上。

```
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 10.2.1.1
```

检查以确认 router2 是否是组 224.1.1.1 的 RP：

```
<#root>
```

```
router2#
```

```
show ip pim rp map
```

```
PIM Group-to-RP Mappings Group(s) 224.0.0.0/4 RP 10.1.1.1 (?), v2v1 Info source: 10.1.1.1 (?), elected
```

224.1.1.1 的 RP 是 10.1.1.1。

检查它是否是 router2 的其中一个接口：

```
<#root>
```

```
router2#
```

```
show ip interface brief
```

```
Interface IP-Address OK? Method Status Protocol Ethernet0/0 10.1.1.2 YES NVRAM up up Ethernet1/0 10.2.
```

因为router2不是RP，所以它一定没有收到此RP加入数据包。检查下游路由器为何向router2发送了Join（加入），但绝不能如此：

```
<#root>
```

```
router3#
```

```
show ip pim rp map
```

```
PIM Group-to-RP Mappings Group(s) 224.0.0.0/4 RP 10.1.1.1 (?), v2v1 Info source: 10.1.1.1 (?), elected
```

可以看见，路由器 3 已静态配置了 RP 信息并指向路由器 2，这是不正确的。这解释了路由器 3 将 RP-Join 发送到路由器 2 的原因。

可能的修正

将 router2 作为组 224.1.1.1 的 RP，或者更改 router3 的设置使其引用正确的 RP 地址。

```
<#root>
```

```
router3#
```

```
show run | i rp
```

```
ip pim rp-address 10.2.1.1 override router3#configure terminal Enter configuration commands, one per line
```

更正router3上的配置后，router3引用了正确的RP地址，错误消息随即停止。

```
<#root>
```

```
router3#
```

```
show ip pim rp map
```

```
PIM Group-to-RP Mappings Group(s) 224.0.0.0/4 RP 10.1.1.1 (?), v2v1 Info source: 10.1.1.1 (?), elected
```

收到重复的多播数据包流

原因 1

当在密集模式下配置了两个路由器时就会收到重复的多播数据包。在密集模式下，设备会定期泛洪流。泛洪后，它会修剪掉不需要流的接口。两个路由器也将经历该确认过程并确定哪一个是转发者。每当计时器关闭时都会发生这种情况，直到此过程完成，两台路由器都会转发数据流。这会导致应用程序收到重复的多播流。

可能的修复方法 1

如果将其中一个路由器配置为多播路由，另一个路由器配置为上游 RP，则可以解决此问题。将其配置为在流进入路由器之前将流转换为稀疏模式。这可以防止重复的数据包抵达应用程序。网络并不负责确保不会有重复的数据包抵达终端主机。应用程序将负责处理重复的数据包并忽略不需要的数据。

原因 2

该问题会在 Cisco Catalyst 6500 交换机中出现，这些交换机配置为出口多播复制模式，可通过移除并重新插入任何板卡 [OIR] 进行触发。在 OIR 后，交换矩阵端口出口 [FPOE] 可能会被错误编程，这可导致数据包被定向到错误的交换矩阵出口通道并被发送到错误的线卡。结果是数据包沿环路返回光纤通道，并且当其在正确的板卡上退出时将重复多次。

```
<#root>
```

```
C6509#
```

```
show mls ip multicast capability
```

```
Current mode of replication is Egress Auto replication mode detection is ON Slot Multicast replication
```

可能的修复方法 2

要解决此问题，可以更改为入口复制模式。在从出口复制模式更改为入口复制模式的过程中会出现流量中断，因为要清除并重新安装快捷方式。

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

将Cisco IOS软件升级到不受Cisco Bug ID [CSCeg28814](#)影响的版本，以永久解决此问题。

注意：只有思科注册用户才能访问思科内部工具和漏洞信息。

原因 3

如果禁用终端主机或服务器上的接收端扩展 (RSS) 设置，也会发生此问题。

可能的修复方法 3

RSS 设置有利于在多个 CPU 间快速传输数据。请在终端主机或服务器上启用 RSS 设置。

多播数据包为什么被丢弃？

原因 1

当组播流量流动时，您可能会看到接口上出现过多的清除和输入数据包丢弃。您可以使用 `show interface` 命令检查流量。

```
<#root>
```

```
Switch#
```

```
show interface gigabitethernet 1/0
```

```
!--- Output suppressed MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec, reliability 255/255, txload 1/255
```

```
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  
438000092206 bytes mcast L3 out Switched: ucast: 4939256 pkt, 790351689 bytes mcast: 0 pkt, 0 bytes 13
```

可能的修复方法 1

对于发现过大流量的接口，可以将 SPT 值设置为无限。

进行如下配置：

```
<#root>
```

```
Switch(config-if)#
```

```
ip pim spt-threshold infinity
```

原因 2

当您在任何接口上使用 `ip igmp join-group <group-name>` 命令时，将进行进程交换。如果组播数据包在任何接口上进行进程交换，它将占用更多 CPU，因为它强制所有数据包到该组的进程交换。您可以运行 `show buffers input-interface` 命令并检查异常大小。

<#root>


Switch#

```
show buffers input-interface gigabitethernet 1/0
```

```
Header DataArea Pool Rcnt Size Link Enc Flags Input Output 437C6EAC 8096AE4 Middl 1 434 7 1 280 Gi1/1
```

可能的修复方法 2

您可以使用 `ip igmp static-group <group-name>` 命令，而非 `ip igmp join-group <group-name>` 命令。

 **注意：**由于以前的问题，您可能会看到大约90%的CPU使用率较高。通过可能的修复方法解决这些问题后，CPU 使用率将下降到正常水平。

相关信息

- [使用CLI工具排除组播网络故障](#)
- [组播快速开始配置指南](#)
-

[启用ECMP组播负载拆分](#)

- [IP 多播技术支持](#)
- [IP 路由协议支持](#)
- [Cisco技术支持和下载](#)

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。