

(1) 调试与验证

例 9-44 显示了 R1 和 R2 的配置。这个配置显示了在接口配置中缺少了异步缺省路由。

例 9-44 确认在 R1 和 R2 的异步接口中缺少异步缺省路由

```
R1#
interface Async1
  description ASYNC LINE TO R2
  ip address 131.108.1.1 255.255.255.0
  encapsulation ppp
  async mode dedicated
  dialer in-band
  dialer map ip 131.108.1.2 name Router2 broadcast
  dialer-group 1
  ppp authentication chap
```

```
R2#
interface Async1
  description ASYNC LINE TO R1
  ip address 131.108.1.2 255.255.255.0
  encapsulation ppp
  async mode dedicated
  dialer in-band
  dialer map ip 131.108.1.1 name Router2 broadcast
  dialer-group 1
  ppp authentication chap
```

(2) 解决方法

在这个例子中, 使用 **async default routing** 或者 **async dynamic routing** 来解决这个问题。

例 9-45 显示了使用 **async default routing** 之后 R1 和 R2 的配置。

例 9-45 配置 R1 和 R2 使用 async default routing

```
R1#
interface Async1
  description ASYNC LINE TO R2
  ip address 131.108.1.1 255.255.255.0
  encapsulation ppp
  async default routing
  async mode dedicated
  dialer in-band
  dialer map ip 131.108.1.2 name Router2 broadcast
  dialer-group 1
  ppp authentication chap
```

```
R2#
interface Async1
  description ASYNC LINE TO R1
  ip address 131.108.1.2 255.255.255.0
```

(待续)

enc
asy
asyn
dia
dia
dia
ppp

例 9-4

例 9-4

R2#s
Neig
131.

13.

这是
为广播类
分组。当
络类型也
组。第 8

图 9-
个问题可

131.1

图 9

```
encapsulation ppp
async default routing
async mode dedicated
dialer in-band
dialer map ip 131.108.1.1 name Router2 broadcast
dialer-group 1
ppp authentication chap
```

例 9-46 显示了在解决这个问题后 OSPF 形成了邻居关系。

例 9-46 确认在使用了 `async default routing` 之后 R1 和 R2 形成了邻居关系

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/-	00:00:32	131.108.1.1	Asyncl

13. OSPF 邻居表为空——原因：在 NBMA 中没有定义网络类型或邻居

这是 NBMA 网络的一个典型问题。除非你配置一个 **neighbor** 语句或者将网络类型改变为广播类型或者点到点类型，否则 OSPF 或者任何其它路由协议将不能发送或接收任何 Hello 分组。当配置了 **neighbor** 语句后，它就触发 OSPF Hello 分组从而邻居关系形成了。改变网络类型也改变了接口行为；在广播网络类型的情况中，OSPF 开始发送和接收 OSPF Hello 分组。第 8 章给出了 OSPF 网络类型的详细说明。

图 9-17 显示了两个路由器在帧中继云中运行 OSPF 的网络图。帧中继只是一个例子；这个问题可能会在任何非广播网络中产生，如 X.25、SMDS 等等。



图 9-17 OSPF 运行在非广播介质上

图 9-18 显示了解决这个问题应遵循的流程图。

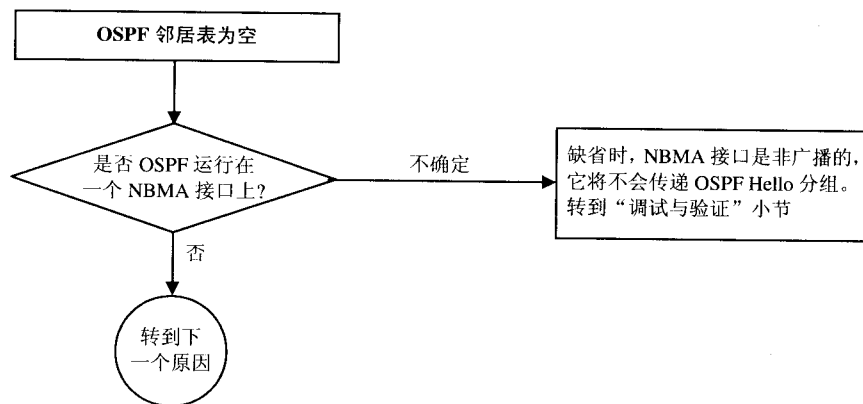


图 9-18 解决问题流程图

(待续)

(1) 调试与验证

例 9-47 显示了 R2 上 **show interface serial0** 的输出。网络类型显示为非广播类型。任何的非广播接口——例如 X.25、SMDS、帧中继等等——总是显示网络类型为非广播类型。

例 9-47 在 R2 的 Serial0 接口确定网络类型

```
R2#show ip ospf interface serial0
Serial0 is up, line protocol is up
Internet Address 131.108.1.2/24, Area 1
Process ID 1, Router ID 131.108.1.2, Network Type NON_BROADCAST, Cost: 64
Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 131.108.1.2, Interface address 131.108.1.2
No backup designated router on this network
Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
Hello due in 00:00:00
Neighbor Count is 0, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)
```

(2) 解决方法

要解决这个问题，需在 **router ospf** 下配置 **neighbor** 语句，如同例 9-48 中所做的一样。通过配置 **neighbor** 语句，OSPF 开始以单播而不是组播的形式发送 Hello 分组。当任何介质的组播能力都被损坏的时候这也是一个有用的技术。确信定义了正确的邻居地址，否则 OSPF Hello 分组将不会使它发送到邻居那儿去。

例 9-48 OSPF 配置了 neighbor 语句

```
R2#
router ospf 1
network 131.108.0.0 0.0.255.255 area 1
neighbor 131.108.1.1 priority 1
```

解决这个问题的其它方法包括改变网络类型为广播类型或者点到多点类型。在这种情况下，OSPF 开始通过链路发送组播 Hello 分组。

例 9-49 显示了如何改变网络类型为广播类型，接着显示了在使用广播网络类型后 **show interface serial0** 的输出。

例 9-49 确认广播网络类型配置允许 OSPF 形成一个邻接体关系

```
R2#
interface Serial 0
ip ospf network broadcast

R2#show ip ospf interface serial0
Serial0 is up, line protocol is up
Internet Address 131.108.1.2, Area 1
Process ID 1, Router ID 131.108.1.2, Network Type BROADCAST, Cost: 64
Transmit Delay is 1 sec, State BDR, Priority 1
Designated Router (ID) 131.108.2.1, Interface address 131.108.1.1
```

(待续)

```
Backup Designated router (ID) 131.108.1.2, Interface address 131.108.1.2
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:05
Neighbor Count is 1, Adjacent neighbor count is 1
Adjacent with neighbor 131.108.2.1 (Designated Router)
Suppress hello for 0 neighbor(s)
```

类似地, 改变网络类型为点到多点也会使它工作。例 9-50 显示了如何改变网络类型为点到多点类型接着显示了 `show ip ospf neighbor` 的输出, 它显示了在作出改变之后邻居通过串行链路进入了 FULL 状态。

例 9-50 确认点到多点网络类型配置允许 OSPF 形成邻接体关系

```
R2#
interface Serial 0
ip ospf network point-to-multipoint
```

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/-	00:01:42	131.108.1.1	Serial0

14. OSPF 邻居表为空——原因: 在两边的帧中继/拨号接口中都缺少关键字 `broadcast`

OSPF 使用组播 Hello 分组来形成邻接体关系。其它的路由协议——例如, RIP 和 EIGRP——也使用广播或者组播来形成邻居关系。在帧中继或拨号接口的情况下, 必须在两端的 `frame-relay` 或者 `dialer-map` 语句中都使用 `broadcast` 关键字来传送 OSPF Hello 分组。这些映射语句仅仅在接口实际上为多点的情况下才有效。例如, 缺省时, 帧中继接口为多点的。BRI 接口也是多点的, 因为它能够拨不止一个号码。

在这要注意的一个事情是两边都必须在 `frame-relay` 或者 `dialer-map` 配置中缺少 `broadcast` 关键字才会产生这个问题。如果仅仅一边缺少 `broadcast` 关键字, 另一边可以发现这个路由器处于 INIT 状态中, 但是邻居永远不会变成邻接体。这种情况在本章后面“问题: OSPF 邻居停滞于 INIT 状态”一节中讨论。

图 9-19 显示了解决这个问题应遵循的流程图。

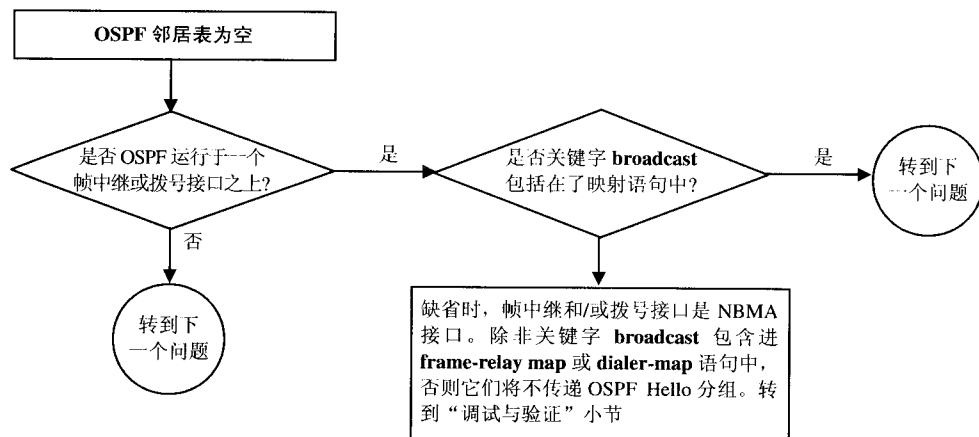


图 9-19 解决问题流程图

(1) 调试与验证

例 9-51 显示了 **debug ip packet 100 detail** 的输出, 表示从 R1 产生的 Hello 分组由于封装错误而没有通过。这儿的访问列表仅仅用于调试目的。这一访问列表监视那些从源地址 131.108.1.1 和 131.108.1.2 而来, 向目的地址 224.0.0.5 发送的 IP 分组。

例 9-51 确认由于封装错误 OSPF Hello 分组被丢弃了

```
R1#show access-list 100
Extended IP access list 100
    permit ip 131.108.1.0 0.0.0.3 host 224.0.0.5 (8 matches)
R1#debug ip packet 100 detail
IP packet debugging is on (detailed) for access list 100
R1#
IP: s=131.108.1.2 (Serial0), d=224.0.0.5, len 64, rcvd 0, proto=89
IP: s=131.108.1.1 (local), d=224.0.0.5 (Serial0), len 68, sending broad/multicast,
proto=
89
IP: s=131.108.1.1 (local), d=224.0.0.5 (Serial0), len 68, encapsulation failed, proto=
89
```

例 9-52 显示了 R1 和 R2 的配置。这个配置显示 **frame-relay map** 语句中缺少关键字 **broadcast**。

例 9-52 R1 和 R2 的配置显示缺少了 broadcast 关键字

```
R1#
interface Serial0
ip address 131.108.1.1 255.255.255.0
encapsulation frame-relay
frame-relay map ip 131.108.1.2 16

R2#
interface Serial0
ip address 131.108.1.2 255.255.255.0
encapsulation frame-relay
frame-relay map ip 131.108.1.1 16
```

(2) 解决方法

例 9-53 显示了 R1 和 R2 修改后的配置, 修改配置后就解决了这个问题。再重复一次, 必须在两边都使用关键字 **broadcast**。如果只在一边使用, 它将产生一个停滞于 INIT 的问题, 这在本章后面讨论。

例 9-53 为 R1 和 R2 的 frame-relay map 语句添加关键字 broadcast

```
R1#
interface Serial0
ip address 131.108.1.1 255.255.255.0
encapsulation frame-relay
ip ospf network broadcast
```

(待续)

fram

```
R2#
inter
ip a
enca
ip o
fram
```

你可以

例 9-5

```
R1#
inter
ip a
enca
dial
```

```
R2#
inter
ip a
enca
diale
```

例 9-55

上形成了。

例 9-55

```
R2#sho
Neigh
131.10
```

9.2.2

这个问题

个路由器试
本身并不是
如果一个路
状态的细节

这个问题

● 错误

● 在 N

图 9-20

于 ATTEMPT

```

frame-relay map ip 131.108.1.2 16 broadcast
R2#
interface Serial0
ip address 131.108.1.2 255.255.255.0
encapsulation frame-relay
ip ospf network broadcast
frame-relay map ip 131.108.1.1 16 broadcast

```

你可以在一个拨号接口使用类似配置，如例 9-54 中所示范的一样。

例 9-54 为 R1 和 R2 的 dialer map 语句添加关键字 broadcast

```

R1#
interface BRI0
ip address 131.108.1.1 255.255.255.0
encapsulation ppp
dialer map ip 131.108.1.2 broadcast name R2 76444
R2#
interface BRI0
ip address 131.108.1.2 255.255.255.0
encapsulation ppp
dialer map ip 131.108.1.1 broadcast name R1 76555

```

例 9-55 显示在解决了这个问题之后一个 OSPF 邻接体在一个使用帧中继封装的串行接口上形成了。

例 9-55 确认 R1 和 R2 的新配置成功了

```

R2#show ip ospf neighbor

```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/DR	00:00:32	131.108.1.1	Serial0

9.2.2 问题：OSPF 邻居停滞于 ATTEMPT 状态

这个问题仅仅在定义了 neighbor 语句的 NBMA 网络中有效。停滞于 ATTEMPT 表示一个路由器试图通过发送它的 Hello 分组联络一个邻居但是没有收到任何回应。ATTEMPT 状态本身并不是一个问题，这因为它是在 NBMA 模式中一个路由器要通过的一个普通状态；但是，如果一个路由器在这个状态停滞很长一段时间，就表明出现了问题。第 8 章讨论了 ATTEMPT 状态的细节。

这个问题最常见的可能原因如下：

- 错误配置了 neighbor 语句。
- 在 NBMA 中单播中断了。

图 9-20 显示了一个网络，其中两个路由器运行 OSPF。这个网络设置用于产生一个停滞于 ATTEMPT 状态的问题。

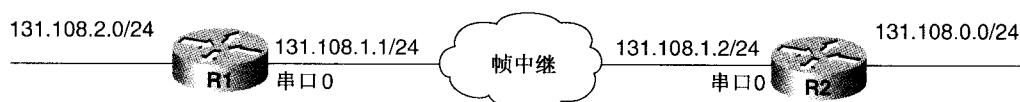


图 9-20 容易产生停滞于 ATTEMPT 状态问题的网络设置

例 9-56 显示了 **show ip ospf neighbor** 的输出，表明邻居停滞于 ATTEMPT 状态。邻居 ID 字段为 N/A，表示这个路由器没关于邻居的任何信息——这就是这个字段为什么是 N/A 的原因；否则的话，它将显示邻居的路由器 ID。

例 9-56 OSPF 邻居停滞于 ATTEMPT 状态

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
N/A	0	ATTEMPT/DROTHER	00:01:57	131.108.1.1	Serial0

1. OSPF 邻居停滞于 ATTEMPT 状态——原因：错误的邻居状态配置

如果 **neighbor** 语句被手工配置在 **router ospf** 配置中，OSPF 将在 NBMA 接口发送一个单播分组。这个 **neighbor** 语句定义了 OSPF 分组的目的 IP 地址。如果 **neighbor** 语句不正确，OSPF 就不能向正确的邻居发送分组。造成一个配置错误很常见，因此，如果过了一會兒邻居还没有出现，就应检查 OSPF 配置中的 **neighbor** 语句或者命令 **show ip ospf neighbor** 输出中的 **neighbor** 语句。如果邻居显示处于 ATTEMPT 状态中，表明这个路由器正试图通过发送 Hello 分组与一个邻居联络，但是它没有从邻居收到任何回应。

图 9-21 显示了解决这个问题应遵循的流程图。

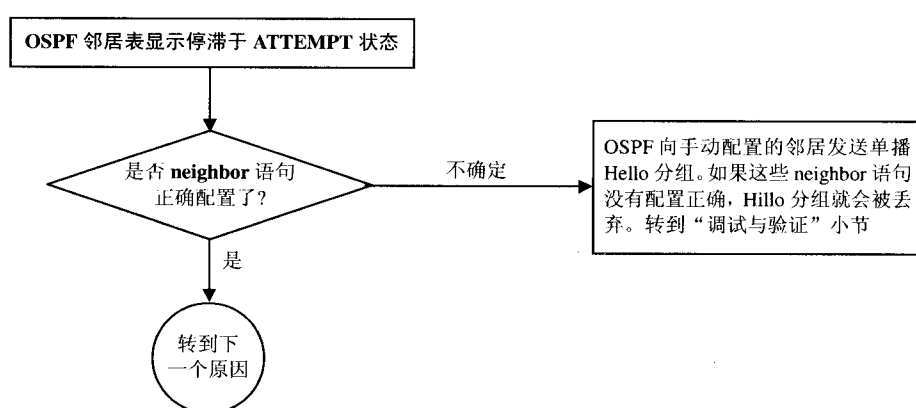


图 9-21 解决问题流程图

(1) 调试与验证

在例 9-57 中，**show ip ospf neighbor** 的输出表示邻居停滞于 ATTEMPT 状态。配置了 **neighbor** 语句，但是邻居的 IP 地址是不正确的。不是 131.108.1.1（如图 9-20 所示），它显示的是 131.108.1.11。

例 9-57 show ip ospf neighbor 命令的输出表明邻居停滞于 ATTEMPT 状态

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
N/A	0	ATTEMPT/DROTHER	00:01:57	131.108.1.11	Serial0

例 9-58 显示了 R2 的配置，表明 **neighbor** 语句也配置错了。

例 9-58 R2 的配置中有一个不正确的 neighbor 语句

```
R2#
```

```
router ospf 1
network 131.108.0.0 0.0.255.255 area 1
neighbor 131.108.1.11 priority 1
```

(2) 解决方法

要解决这个问题，需给相应的 **neighbor** 语句配置相应的 IP 地址。例 9-59 显示了解决这个问题的 R2 的新配置。

例 9-59 在 R2 上配置正确的 neighbor 语句以解决这个问题

```
R2#
```

```
router ospf 1
network 131.108.0.0 0.0.255.255 area 1
neighbor 131.108.1.1 priority 1
```

例 9-60 显示了解决这个问题后 **show ip ospf neighbor** 的输出。

例 9-60 确认新的 neighbor 语句解决了这个问题

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/DR	00:01:42	131.108.1.1	Serial0

2. OSPF 邻居停滞于 ATTEMPT 状态——原因：在 NBMA 中单播连接中断了

如果 **neighbor** 语句手工配置好了，OSPF 将在 NBMA 接口发送单播 Hello 分组。如果单播连接中断了，OSPF 将永远不会形成任何邻接体关系。在 NBMA 接口上，OSPF 缺省等待每 Hello 间隔（也就是每 30 秒）就尝试与邻居联络一次。如果它没有从邻居收到任何回答，将表示邻居停滞于 ATTEMPT 状态。对于单播连接中断存在很多可能的原因。假设第 2 层是连接好的，你应该考虑下列单播连接中断的原因：

- 在一个帧中继或者 ATM 交换机中存在一个错误的 DLCI 或者 VPI / VCI 映射。
- 一个访问列表阻止单播。
- NAT（网络地址转换）转换了单播。

图 9-22 显示了解决这个问题应遵循的流程图。

(1) 调试与验证

例 9-61 显示了从 R2 发送到 R1 的 ping 的输出。Ping 显示 100% 失败。因为 **ping** 命令使

用了 ICMP 分组并且它是一个单播分组，失败表明单播连接中断了。

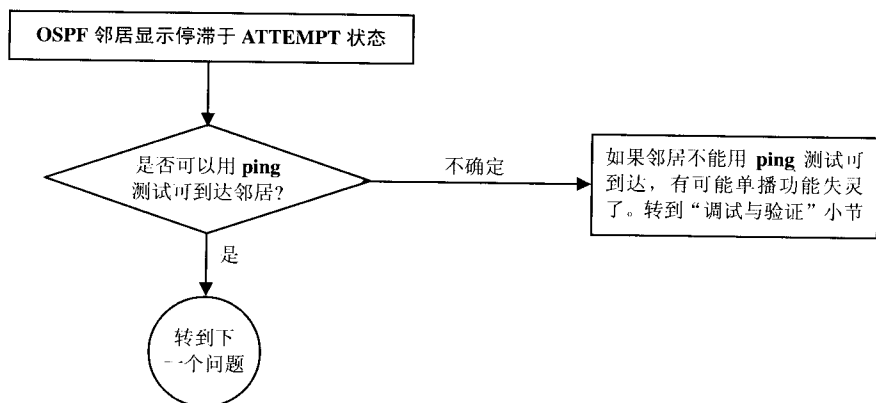


图 9-22 解决问题流程图

例 9-61 ping 失败表示存在一个连接问题

```

R2#ping 131.108.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 131.108.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R2#
  
```

(2) 解决方法

如同前面提到的，单播中断连接可能是很多因素的结果。如果是一个错误的 DLCI 或者 VC 映射，确信检查这些映射并纠正它们。如果有一个访问列表阻止单播连接，确信在访问列表中允许所需的单播 IP 地址。例 9-62 显示了解决这个单播连接问题之后 `show ip ospf neighbor` 命令的输出。

例 9-62 确认单播可重新运行了并且 OSPF 形成了邻居关系

```

R2#show ip ospf neighbor

Neighbor ID    Pri  State       Dead Time   Address        Interface
131.108.2.1    1    FULL/DR     00:01:42    131.108.1.1    Serial0
  
```

9.2.3 问题：OSPF 邻居停滞于 INIT 状态

当一个路由器从邻居收到一个 OSPF Hello 分组的时候，它在 Hello 分组中包含进邻居的路由器 ID 并发送这个 Hello 分组。如果它不包含邻居的路由器 ID，那么邻居将停滞于 INIT 状态，这表明出现了一个问题。路由器收到的第一个分组将使路由器进入 INIT 状态。这并不是问题，但是如果路由器在这个状态停留很长一段时间，就表明出现了问题。它表示邻居路由器没有看到由这个路由器发送的 Hello 分组——那也就是为什么它没能包括这个路

由器的路
滞于 INIT

这个

●

●

●

●

●

例 9-

例 9-

R2#s

Neig

131.

1. O

OSPF

义了一个

确地允许；

有一边阻

的输出返

图 9-2

(1) 调

例 9-64

访问列表 10

由器的路由器 ID 在它自己的 Hello 分组中的原因。图 9-20 中的网络设置在这里用来讨论停滞于 INIT 状态的问题。

这个问题最常见的可能原因如下：

- 访问列表在某一端阻塞 OSPF Hello 分组。
- 组播功能在某一端被破坏了（6500 交换机问题）。
- 验证只在在一端可用（虚链路例子）。
- 在某一端的 **fame-relay map / dialer map** 语句中缺少 **broadcast** 关键字。
- Hello 分组在某一端的第 2 层丢失了。

例 9-63 显示了 **show ip ospf neighbor** 命令的输出，表明 R2 的邻居停滞于 INIT 状态。

例 9-63 **show ip ospf neighbor** 命令输出表明 R2 的邻居停滞于 INIT 状态

R2#show ip ospf neighbor					
Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	INIT/-	00:00:33	131.108.1.1	Ethernet0

1. OSPF 邻居停滞于 INIT 状态——原因：访问列表在某一端阻止 OSPF Hello 分组

OSPF 使用一个组播地址 224.0.0.5 用于发送和接收 Hello 分组。分组如果在一个接口定义了一个访问列表并且在那个接口上启用了 OSPF，那么这个组播地址必须在访问列表中明确地允许；否则，它可能产生如停滞于 INIT 状态等问题。停滞于 INIT 状态的问题仅仅在只有一端阻止 OSPF 的时候发生。如果两边都阻止 OSPF Hello 分组，则 **show ip ospf neighbor** 的输出返回一个空表。

图 9-23 显示了解决这个问题应遵循的流程图。

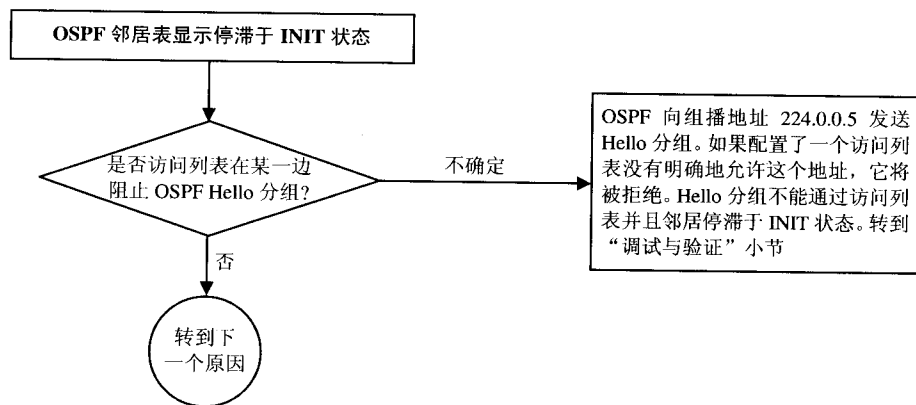


图 9-23 解决问题流程图

(1) 调试与验证

例 9-64 显示了 R1 上 **show access-list 101** 和 **debug ip packet 101 detail** 的输出，配置了访问列表 101 所以仅仅可以看见在 R1 和 R2 之间的 OSPF Hello 分组。

例 9-64 调试的输出显示 OSPF Hello 分组被拒绝

```

R1#show access-list 101
Extended IP access list 101
    permit ip 131.108.1.0 0.0.0.3 host 224.0.0.5 (8 matches)
R1#debug ip packet 101 detail
IP packet debugging is on (detailed) for access list 101
R1#
IP: s=131.108.1.1 (local), d=224.0.0.5 (Ethernet0), len 60, sending broad/multicast,
proto
=89
IP: s=131.108.1.2 (Ethernet0), d=224.0.0.5, len 82, access denied, proto=89
IP: s=131.108.1.1 (local), d=224.0.0.5 (Ethernet0), len 60, sending broad/multicast,
proto
=89
IP: s=131.108.1.2 (Ethernet0), d=224.0.0.5, len 82, access denied, proto=89

```

例 9-65 显示了 R1 的配置。在 R1 上的访问列表 100 仅允许目的地址为 R1 和 R2 接口地址的流量；它拒绝任何其它流量，包括 OSPF Hello 分组。在路由器 R1 上的访问列表 101 被配置限制调试因此它将仅仅显示通过的 OSPF Hello 分组。

例 9-65 在 R1 上配置的访问列表阻止 OSPF Hello 分组

```

R1#
!
interface Ethernet0
 ip address 131.108.1.1 255.255.255.0
 ip access-group 100 in
!
access-list 100 permit ip any 131.108.1.0 0.0.0.255

```

(2) 解决方法

要解决这个问题，需在 R1 上的访问列表 100 中允许 OSPF Hello 分组。新的一行允许任何源地址为 131.108.10-255、目的地址为 OSPF 组播地址 224.0.0.5 的分组。例 9-66 显示了 R1 上已修改的访问列表。

例 9-66 R1 上已修改的访问列表

```

R1#
access-list 100 permit ip any 131.108.1.0 0.0.0.255
access-list 100 permit ip 131.108.1.0 0.0.0.255 host 224.0.0.5

```

例 9-67 显示了解决这个问题后 **show ip ospf neighbor** 的输出。

例 9-67 **show ip ospf neighbor** 命令的输出确认访问列表现在允许 OSPF 组播并且 OSPF 邻居形成了

```

R2#show ip ospf neighbor

```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/DR	00:00:39	131.108.1.1	thernet0

2. OSPF 邻居停滞于 INIT 状态——原因：组播功能在某一端被破坏了（6500 交换机问题）

这是一个特殊情况，仅在带有多层交换功能卡（MSFC）的 Catalyst 6500 交换机的情况下才会发生。问题是一边发送 OSPF Hello 分组但另一边没有收到。在图 9-24 中的网络设置显示了这将会如何成为一个问题。



图 9-24 因为组播功能损坏了而容易导致 OSPF 邻居停滞于 INIT 状态的网络设置

当在 6500 交换机上键入命令 `set protocolfilter enabled` 时会产生这种情况。缺省时，协议过滤器是不可用的。使这个命令可用开始在 6500 交换机的 Flex Wan 模块中改变发往或者由 MSFC 和端口适配器传来的组播帧。图 9-25 显示了解决这个问题遵循的流程图。

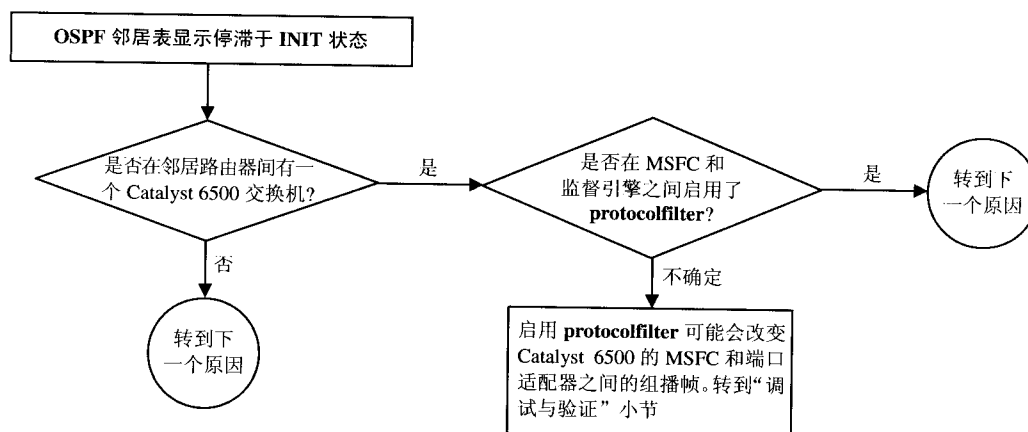


图 9-25 解决问题流程图

(1) 调试与验证

例 9-68 显示了 `show ip ospf neighbor` 的输出。邻居停滞于 INIT 状态，并且在中间的交换机是带有 MSFC 的 6500，如图 9-24 所示。

例 9-68 OSPF 邻居停滞于 INIT 状态

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1 x	INIT/-	00:00:33	131.108.1.1	FastEthernet0/0

(2) 解决方法

要解决这个问题，需像下面一样禁用 6500 交换机上的协议过滤器：

```
CAT6K(enable) set protocolfilter disable
```

例 9-69 显示了解决这个问题之后 OSPF 邻居处于 FULL 状态。

例 9-69 确认在 6500 交换机上的协议过滤器关掉之后 OSPF 邻居完好了

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/DR	00:00:33	131.108.1.1	FastEthernet0/0

3. OSPF 邻居停滞了 INIT 状态——原因：验证只在一边启用

使用验证时，就必须在两边都启用；否则一边会显示邻居停滞于 INIT 状态。使用了验证，路由器将拒绝所有非验证分组，并且邻接体状态将显示停滞于 INIT 状态。另一边将不会检测到任何问题，因为打开了验证，因此它将简单地忽略分组中的验证并将它看作一个普通分组。

图 9-26 显示了解决这个问题应遵循的流程图。

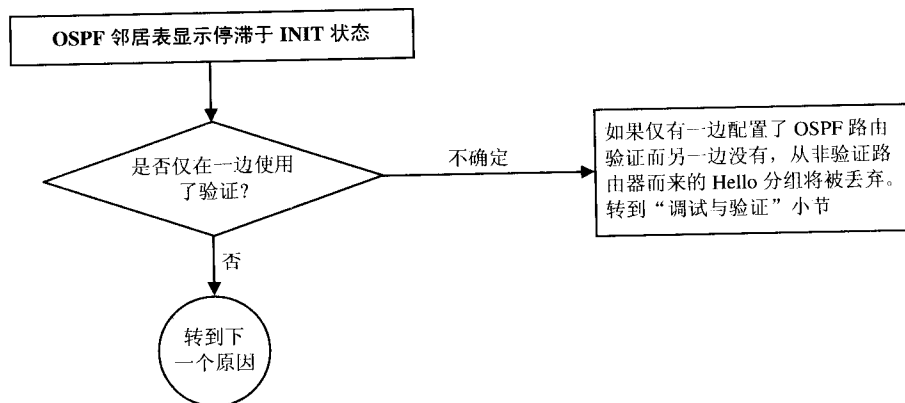


图 9-26 解决问题流程图

(1) 调试与验证

例 9-70 显示了 R2 上 `debug ip ospf adj` 的输出，表明路由器 R2 使用了纯文本验证但是 R1 发送没有任何验证的分组。因此，R2 会拒绝这些分组。这是一个纯文本验证的例子。在 MD5 验证的情况下，调试的输出将显示 `we use type 2`。

例 9-70 debug ip ospf adj 命令输出表明在邻居路由器有一个不匹配的验证类型

```
R2#debug ip ospf adj
```

```
OSPF adjacency events debugging is on
```

```
R2#
```

```
OSPF: Rcv pkt from 131.108.1.1, Ethernet0 : Mismatch Authentication type. Input packet specified type 0, we use type 1
```

例 9-71 显示了 R2 的配置。配置显示 R2 在区域 1 中使用了纯文本验证。无论是否在这个接口上定义了验证密钥，这个问题都会重复出现。如果没有定义密钥，路由器将使用一个缺省密钥。

例 9-7

```
R2#
route
netw
are
```

(2) 角

要解决

的新配置，

例 9-7

```
R2#
!
inter
ip a
ip c
!
route
netw
are
```

```
R1#
!
inter
ip a
ip c
!
route
netw
are
```

例 9-7

例 9-7

```
R2#s
Neig
131
```

相似的

误就是没有

成了一个

工配置在

4. O

句中缺少

OSPF

送或接收

例 9-71 R2 在区域 1 使用纯文本验证

```
R2#
router ospf 1
 network 131.108.1.0 0.0.0.255 area 1
 area 1 authentication
```

(2) 解决方法

要解决这个问题，需在两边使用验证并在两边定义验证密钥。例 9-72 显示了 R1 和 R2 的新配置，更改这些配置后解决了问题。

例 9-72 在两个路由器上配置验证来解决这个问题

```
R2#
!
interface Ethernet0
 ip address 131.108.1.2 255.255.255.0
 ip ospf authentication-key cisco
!
router ospf 1
 network 131.108.1.0 0.0.0.255 area 1
 area 1 authentication
```

```
R1#
!
interface Ethernet0
 ip address 131.108.1.1 255.255.255.0
 ip ospf authentication-key cisco
!
router ospf 1
 network 131.108.1.0 0.0.0.255 area 1
 area 1 authentication
```

例 9-73 显示了解决这个问题后的邻居状态。

例 9-73 show ip ospf neighbor 命令输出确认修改后的验证解决了这个问题

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/DR	00:00:33	131.108.1.1	Ethernet0

相似的问题发生在有一条虚链路的情况中。当在主干路由器上使用验证时，很常见的错误就是没有在连接两个不同区域的路由器上使用验证。这个路由器在创建一条虚链路之后变成了一个虚 ABR；因此，必须在那个路由器上为区域 0 启用验证功能，尽管区域 0 并没有手工配置在该路由器之上。

4. OSPF 邻居停滞于 INIT 状态——原因：在某一边的 fame-relay map/dialer map 语句中缺少 broadcast 关键字

OSPF 使用了一个组播地址 224.0.0.5 发送和接收 OSPF Hello 分组，如果一边不能发送或接收 Hello 分组，OSPF 邻居将停滞于 INIT 状态。这儿要注意的重要的事情就是仅仅

只有一边受困于这个组播问题。R1 看到邻居处于 INIT 状态中但是也还可以看到邻居的 Hello 分组而没有任何问题。当 R1 向 R2 发送 Hello 分组时, 它将不能到达 R2 因为第 2 层不能发送任何广播或组播分组。这因为在 R1 的 **frame-relay map** 语句中缺少 **broadcast** 关键字。在 ISDN 或者拨号接口中当 **dialer map** 语句没有配置关键字 **broadcast** 的时候也会发生相似的问题。

图 9-27 显示了用于讨论这个问题的网络设置。

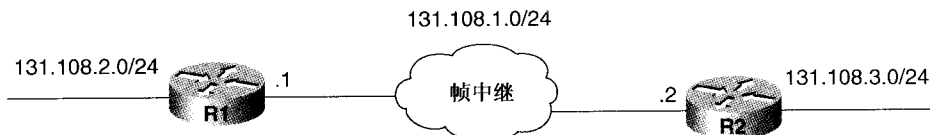


图 9-27 用于产生 OSPF 邻居停滞于 INIT 状态问题的网络拓扑

图 9-28 显示了解决这个问题应遵循的流程图。

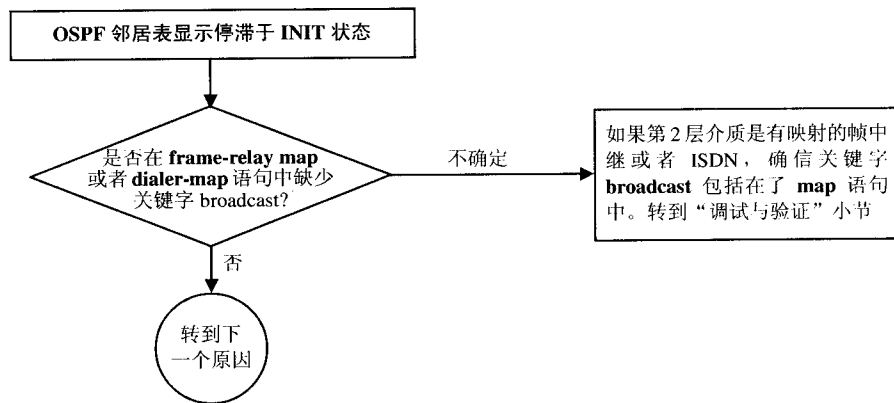


图 9-28 解决问题流程图

(1) 调试与验证

例 9-74 中 **debug ip packet 100 detail** 的输出表明由 R1 产生的 Hello 分组由于一个封装的失败而导致不能穿越。

例 9-74 封装失败阻止 Hello 分组从 R1 向外传播

```
R1#show access-list 100
Extended IP access list 100
    permit ip 131.108.1.0 0.0.0.3 host 224.0.0.5 (8 matches)
R1#debug ip packet 100 detail
IP packet debugging is on (detailed) for access list 100
R1#
IP: s=131.108.1.2 (Serial0), d=224.0.0.5, len 64, rcvd 0, proto=89
IP: s=131.108.1.1 (local), d=224.0.0.5 (Serial0), len 68, sending broad/multicast,
    proto=89
IP: s=131.108.1.1 (local), d=224.0.0.5 (Serial0), len 68, encapsulation failed,
    proto=89
```

例 9-7
字 broadcast

例 9-7

```
R1#
inter
ip a
enca
fram
R2#
inter
ip a
enca
fram
```

(2) 解

要解决

broadcast.

例 9-7

```
R1#
inter
ip a
enca
ip o
fram
R2#
inter
ip a
enca
ip o
fram
```

例 9-7

例 9-7

```
R2#sh
Neigh
131.1
```

5. OS

当第 2

组播流量。

组, R2 的

例 9-75 显示了 R1 和 R2 的配置。配置显示在 R1 上的 **frame-relay map** 语句中缺少关键字 **broadcast**。可是 R2 却有着正确的 **frame-relay map** 语句。

例 9-75 R1 和 R2 的配置, R1 忽略了关键字 **broadcast**

```
R1#
interface Serial0
 ip address 131.108.1.1 255.255.255.0
 encapsulation frame-relay
 frame-relay map ip 131.108.1.2 16

R2#
interface Serial0
 ip address 131.108.1.2 255.255.255.0
 encapsulation frame-relay
 frame-relay map ip 131.108.1.1 16 broadcast
```

(2) 解决方法

要解决这个问题, 需确信在 **frame-relay map** 和 **dialer-map** 语句中配置了关键字 **broadcast**。例 9-76 显示了 R1 和 R2 的新配置, 以解决这个问题。

例 9-76 在 R1 上改正 frame-relay map 语句以包括关键字 **broadcast**

```
R1#
interface Serial0
 ip address 131.108.1.1 255.255.255.0
 encapsulation frame-relay
 ip ospf network broadcast
 frame-relay map ip 131.108.1.2 16 broadcast

R2#
interface Serial0
 ip address 131.108.1.2 255.255.255.0
 encapsulation frame-relay
 ip ospf network broadcast
 frame-relay map ip 131.108.1.1 16 broadcast
```

例 9-77 显示了在解决这个问题后使用帧中继封装通过串行接口形成了 OSPF 邻接体关系。

例 9-77 show ip ospf neighbor 命令输出表明问题解决了

```
R2#show ip ospf neighbor

Neighbor ID  Pri  State           Dead Time   Address        Interface
131.108.2.1  1    FULL/BDR       00:00:32   131.108.1.1   Serial0
```

5. OSPF 邻居停滞于 INIT 状态——原因: Hello 分组在某一边的第 2 层丢失了

当第 2 层介质存在问题的时候会发生这种情况, 例如, 帧中继交换机由于某些原因阻止组播流量。当 R1 发送 Hello 分组时, R2 将永远收不到。因为 R2 看不到从 R1 来的 Hello 分组, R2 的邻居表将为空。可是, R1 看到了从 R2 来的 Hello 分组, 但分组中并没有将 R1 作

为一个可用的邻居而列出；因此，R1 宣布这个邻居处于 INIT 状态。

图 9-29 显示了解决这个问题应遵循的流程图。

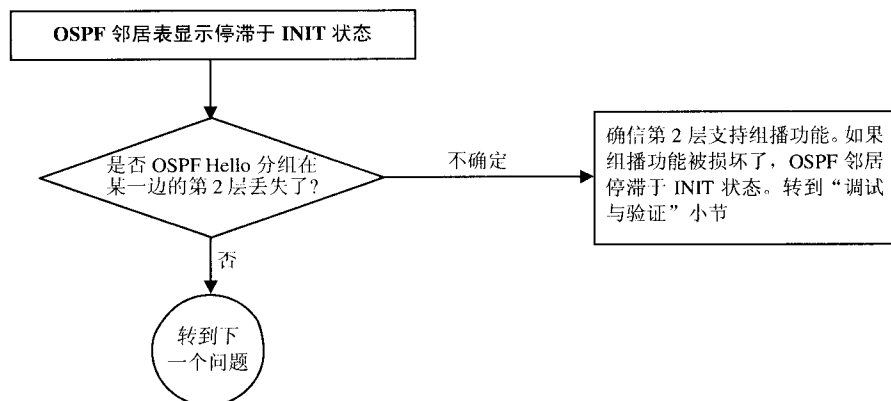


图 9-29 解决问题流程图

(1) 调试与验证

例 9-78 显示了在 R1 和 R2 上 **debug ip packet detail** 的输出。这个调试打开了访问列表 100，它显示 R1 发送和接收 OSPF Hello 分组，但是 R2 仅仅发送而没有接收到任何 OSPF Hello 分组。

例 9-78 debug 输出显示 R2 发送但是没有从 R1 接收任何 Hello 分组

```

R1#show access-list 100
Extended IP access list 100
    permit ip 131.108.1.0 0.0.0.3 host 224.0.0.5 (8 matches)
R1#debug ip packet 100 detail
IP packet debugging is on (detailed) for access list 100
R1#
IP: s=131.108.1.2 (Serial0), d=224.0.0.5, len 64, rcvd 0, proto=89
IP: s=131.108.1.1 (local), d=224.0.0.5 (Serial0), len 68, sending broad/multicast,
    proto=89

R2#show access-list 100
Extended IP access list 100
    permit ip 131.108.1.0 0.0.0.3 host 224.0.0.5 (8 matches)
R2#debug ip packet 100 detail
IP packet debugging is on (detailed) for access list 100
R2#
IP: s=131.108.1.1 (local), d=224.0.0.5 (Serial0), len 68, sending broad/multicast,
    proto=89
IP: s=131.108.1.1 (local), d=224.0.0.5 (Serial0), len 68, sending broad/multicast,
    proto=89
  
```

R2 持续发送 OSPF Hello 分组，但永远收不到从 R1 发送过来的任何 Hello 分组。这表示 R1 的 Hello 分组在中途丢失了，因为调试显示 R1 接收的同时也发送 OSPF Hello 分组。

(2) 解决方法

在两边都进行了调试，很清楚两边都发送 Hello 分组但是 R1 的 Hello 分组没有能传送过去。很可能是帧中继云或者其它的第 2 层介质丢失了这个组播分组。这也可以通过在线路上

使用一个嗅探器。

这个问题。

可行的解决方法。

第 1 步

第 2 步

例 9-79

基本上，接收

neighbor 语

坏的情况下。

例 9-79

```

R1#
interf
ip ad
encap
ip os
frame
R2#
interf
ip ad
encap
ip os
frame
  
```

例 9-80

例 9-80

```

R1#
router
netwo
neig
  
```

这个解
变网络类型

Hello 分组。

配置一个 n

例 9-8

邻接体关系

例 9-8

滞于 INIT 状

```

R2#sh
Neigh
131.1
  
```

使用一个嗅探器 (sniffer) 来确认。

这个问题的解决方法是修复第 2 层组播功能, 这已超出了本书范围。在这种情况下一个可行的解决方法有以下步骤:

第 1 步 在两边都将网络类型改变为非广播类型。

第 2 步 在一个路由器上配置 **neighbor** 语句。

例 9-79 显示了使用的新的接口配置, 这样一个 **neighbor** 语句就可以解决这个问题了。基本上, 接口已经被定义成了非广播类型, 因此可以定义一个 **neighbor** 语句。当定义了一个 **neighbor** 语句之后, OSPF 发送一个单播 Hello 分组。这个配置在任何第 2 层介质组播功能损坏的情况下总能工作。

例 9-79 在两边都将网络类型改变为非广播类型

```
R1#
interface Serial0
ip address 131.108.1.1 255.255.255.0
encapsulation frame-relay
ip ospf network non-broadcast
frame-relay map ip 131.108.1.2 16 broadcast
R2#
interface Serial0
ip address 131.108.1.2 255.255.255.0
encapsulation frame-relay
ip ospf network non-broadcast
frame-relay map ip 131.108.1.1 16 broadcast
```

例 9-80 显示了配置了 **neighbor** 语句以使 OSPF 发送单播 Hello 分组的 OSPF 配置。

例 9-80 配置 **neighbor** 语句以使 OSPF 发送一个单播 Hello 分组

```
R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 1
neighbor 131.108.1.2
```

这个解决方法是一个解决第 2 层问题的方法, 但它不能解决最初的第 2 层问题。通过改变网络类型为非广播类型, 如例 9-79 中所做的, OSPF 将作为单播而不是组播来发送和接收 Hello 分组。因此, 如果第 2 层有任何有关组播的问题发生, 改变网络类型为非广播类型并且配置一个 **neighbor** 语句将使得 OSPF 在一个组播功能损坏的介质之上形成邻居关系。

例 9-81 显示使用了 **neighbor** 命令和一个非广播网络类型后, 通过串行接口形成了 OSPF 邻接体关系。

例 9-81 确认使用一个非广播网络类型解决了由一个第 2 层问题引起的 OSPF 邻居停滞于 INIT 状态的问题

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/DR	00:00:32	131.108.1.1	Serial0

9.2.4 问题：OSPF 邻居停滞于 2-way 状态——原因：在所有路由器上都配置了优先级 0

在广播介质中有一个 2-way 状态是很常见的，因为并非每个路由器都在广播介质中变成了邻接体。每个路由器都与 DR 和 BDR 路由器形成 FULL 状态关系。

在这个例子中，仅有两个路由器在以太网上，且两个都配置了优先级 0。优先级 0 表示这个路由器将不参加 DR / BDR 选举过程。当这个网段中有“低端”路由器并且你想要这些低端路由器不会变为 DR 的时候，这个配置有用。为了达到这个目的，你应该配置优先级 0。缺省时，优先级设置为 1。网段中具有最高优先级的路由器赢得一个 DR 选举。如果所有的优先级都保持缺省值，那么具有最高路由器 ID 的路由器变成 DR。关于 DR 和 BDR 选举的更多信息，请参考第 8 章。

如果一个以太网段中所有的路由器都配置有优先级 0，那么这个网段上将不会有路由器与其它任何路由器进入 FULL 状态。这就产生了问题。在这个网段中，至少必须有一个路由器的优先级不为 0。

图 9-30 显示了受困于这个问题的网络设置。



图 9-30 用于产生 OSPF 邻居停滞于 2-way 状态问题的网络设置

图 9-31 显示了解决这个问题应遵循的流程图。

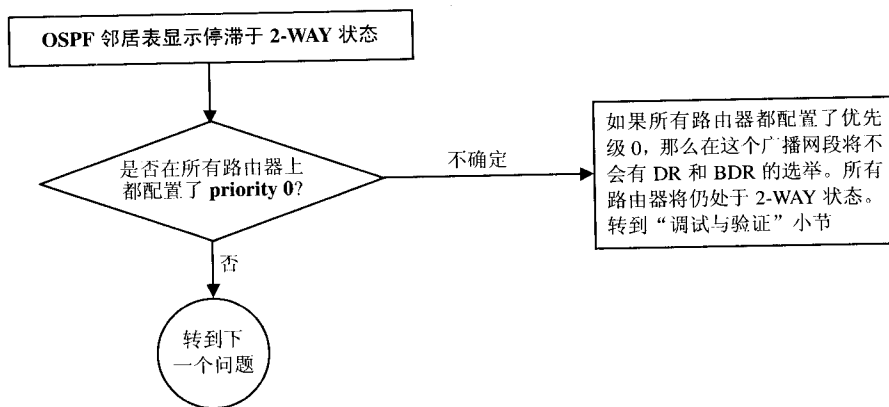


图 9-31 解决问题流程图

(1) 调试与验证

例 9-82 显示了 `show ip ospf neighbor` 的输出。在这个接口没有邻居之间处于 FULL 状态。

例 9-82 `show ip ospf neighbor` 命令输出确定邻居之间处于 2-way 状态

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	0	2-WAY/DR OTHER	00:00:32	131.108.1.1	Ethernet0

例 9-83 显示

例 9-83

```
R1#
interface
ip address
ip ospf

R2#
interface
ip address
ip ospf
```

(2) 解决

要解决这

并且形成一个

例 9-84

```
R1#
interface
ip address
no ip o
```

例 9-85 显

了一个邻接体

例 9-85

```
R2#show
Neighbor
131.108.
```

9.2.5

这在 OSPF
和一个从设备
于 EXSTART
EXCHANGE

引起这个

- 不匹
- 在邻
- 不能
- 由于

例 9-83 显示了 R1 和 R2 的以太网接口都配置了优先级 0。

例 9-83 R1 和 R2r 的 Ethernet 0 接口的优先级设置

```
R1#  
interface Ethernet0  
ip address 131.108.1.1 255.255.255.0  
ip ospf priority 0  
  
R2#  
interface Ethernet0  
ip address 131.108.1.1 255.255.255.0  
ip ospf priority 0
```

(2) 解决方法

要解决这个问题，至少需在一个路由器上去掉 **priority 0** 命令使得路由器变成一个 DR 并且形成一个 FULL 邻接体。例 9-84 显示了 R1 上的配置改变以解决这个问题。

例 9-84 从 R1 上去掉优先级 0 使得它与 R1 形成 FULL 邻接体关系

```
R1#  
interface Ethernet0  
ip address 131.108.1.1 255.255.255.0  
no ip ospf priority 0
```

例 9-85 显示了在 R1 上去掉了 **priority 0** 命令后，问题解决了并且 OSPF 与它的邻居形成了一个邻接体关系。

例 9-85 确认在 R1 上去掉 priority 0 解决了这个问题

```
R2#show ip ospf neighbor  
  
Neighbor ID    Pri   State       Dead Time   Address        Interface  
131.108.2.1    1     FULL/DR     00:00:32   131.108.1.1   Ethernet0
```

9.2.5 问题：OSPF 邻居停滞于 EXSTART / EXCHANGE 状态

这在 OSPF 邻接体过程中是一个重要的状态。在这个状态中，路由器选择了一个主设备和一个从设备还有一个初始序列号。整个的数据库也在这个状态下交换。如果一个邻居停滞于 EXSTART / EXCHANGE 状态持续很长一段时间，那就表明出了问题。关于 EXSTART / EXCHANGE 状态的更多信息请参考第 8 章。

引起这个问题最常见的可能原因如下：

- 不匹配的接口 MTU。
- 在邻居上有重复的路由器 ID。
- 不能用超过一定大小的 MTU 去 ping 通。
- 由于下列原因单播连接损坏了：
 - 在帧中继 / ATM 交换机中错误的 VC / DLCI 映射。

——访问列表阻止单播。

——对单播进行了 NAT 转换。

- 在 PRI 和 BRI / 拨号接口之间的网络类型为点到点网络类型。

图 9-32 显示了两个运行 OSPF 的路由器。这一网络设置在 OSPF 中产生停滞于 EXSTART / EXCHANGE 状态的问题。



图 9-32 产生停滞于 EXSTART / EXCHANGE 状态问题的网络设置

例 9-86 显示了 `show ip ospf neighbor` 的输出，它表示邻居停滞于 EXSTART / EXCHANGE 状态。

例 9-86 `show ip ospf neighbor` 命令输出表示一个邻居停滞于 EXSTART / EXCHANGE 状态

R2#show ip ospf neighbor					
Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	EXSTART/-	00:00:33	131.108.1.1	Serial0

1. OSPF 邻居停滞于 EXSTART / EXCHANGE 状态——原因：不匹配的接口 MTU

OSPF 在一个数据库描述分组中发送接口 MTU。如果有一个 MTU 不匹配，OSPF 将不会形成一个邻接体关系。接口 MTU 选项在 RFC 2178 中加入。以前，没有机制用于检测接口 MTU 不匹配问题。这个选项加入到了 Cisco IOS 软件版 12.0.3 及后续版本中。

图 9-33 显示了解决这个问题应遵循的流程图。

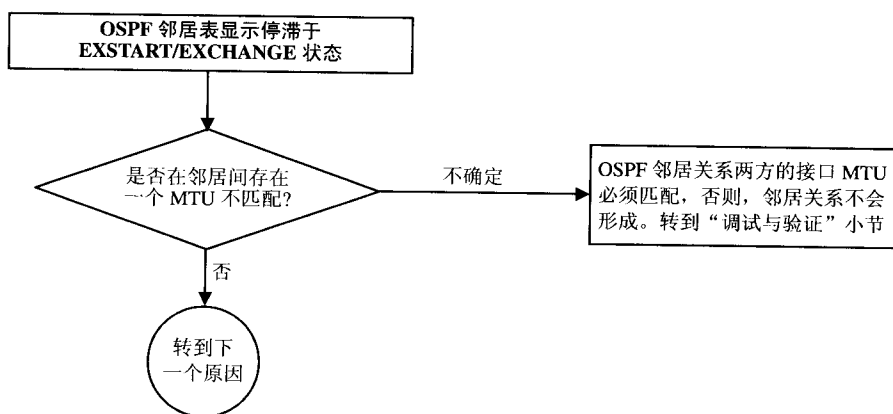


图 9-33 解决问题流程图

(1) 调试与验证

例 9-87 显示了 R1 上 `debug ip ospf adj` 命令的输出，它表示邻居 MTU 大一些。因此，OSPF 不能形成一个邻接体。

例 9-87

```
R1#debug
OSPF: R
OSPF: S
OSPF: R
1500
state E
OSPF: N
```

例 9-88

1400 字节，在

例 9-88

```
R1#show
Serial0
Intern
Broadc
MTU is
```

```
R2#show
Serial0
Intern
Broadc
MTU is
```

(2) 解决

在 Cisco

将在一个 debu

消息不会产生

出现在调试信

要纠正这

配置以解决这

例 9-89

```
R1#
interfac
ip addre
mtu 1500
```

有另一种

嵌有路由交换

例 9-87 debug ip ospf adj 命令输出表示一个不匹配的接口 MTU

```

R1#debug ip ospf adj
OSPF: Retransmitting DBD to 131.108.1.2 on Serial0.1
OSPF: Send DBD to 131.108.1.2 on Serial0.1 seq 0x1E55 opt 0x2 flag 0x7 len 32
OSPF: Rcv DBD from 131.108.1.2 on Serial0.1 seq 0x22AB opt 0x2 flag 0x7 len 32 mtu
1500
state EXSTART
OSPF: Nbr 131.108.1.2 has larger interface MTU

```

例 9-88 显示了在 R1 和 R2 上 **show ip interface** 的输出。R1 上的 IP 接口 MTU 被设置为 1400 字节，在 R2 上，设置为 1500 字节。这就导致了一个 MTU 不匹配问题。

例 9-88 在 R1 和 R2 上的 show ip interface 命令输出准确指出了 MTU 不匹配

```

R1#show ip interface serial 0.1
Serial0.1 is up, line protocol is up
Internet address is 131.108.1.1/24
Broadcast address is 255.255.255.255
MTU is 1400 bytes

```

```

R2#show ip interface serial 0.1
Serial0.1 is up, line protocol is up
Internet address is 131.108.1.2/24
Broadcast address is 255.255.255.255
MTU is 1500 bytes

```

(2) 解决方法

在 Cisco IOS 软件版本 12.0.3 和后续版本中，如果有一个 MTU 不匹配，Cisco IOS 软件将在一个 **debug** 消息中表明这个问题，如例 9-87 中所示。如果 R2 的 MTU 比 R1 的小，这个消息不会产生。如果 R1 没有运行 Cisco IOS 软件版本 12.0.3 或者后续版本，这个消息也不会出现在调试信息中。检测这个 MTU 不匹配的唯一方法就是在两边都检查接口配置。

要纠正这个问题，需确信 MTU 在两边都设置成了一样的值。例 9-89 显示了 R1 上的新配置以解决这个问题。

例 9-89 在 R1 上设置同样的 MTU 值

```

R1#
interface Serial0.1 multipoint
ip address 141.108.10.3 255.255.255.248
mtu 1500

```

有另一种可能导致一个 MTU 不匹配的情况——当一个路由器通过 FDDI 连接到一个内嵌有路由交换模块（RSM）的交换机的时候。图 9-34 显示了这个设置。

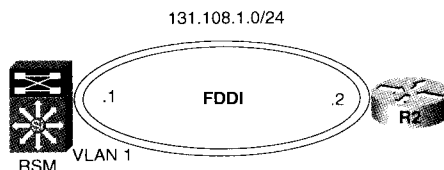


图 9-34 产生 MTU 不匹配问题的网络设置

VLAN 1 接口是一个有着 1500 字节 MTU 的虚拟以太网接口，而在 R2 上的 FDDI 接口的 MTU 为 4470 字节，如例 9-90 所示。

例 9-90 RSM 和 R2 的配置显示 MTU 不匹配

```
RSM#show interface vlan 1
Vlan1 is up, line protocol is up
  Hardware is Cat5k RP Virtual Ethernet, address is 0030.f2c9.8338 (bia 0030.f2)
  Internet address is 131.108.1.1/24
  MTU 1500 bytes , BW 10000 Kbit, DLY 1000 usec,
R2#show interface fddi 0
Fddi0 is up, line protocol is up
  Hardware is DAS FDDI, address is 0000.0c17.acbf (bia 0000.0c17.acbf)
  Internet address is 131.108.1.2/24
  MTU 4470 bytes , BW 100000 Kbit, DLY 100 usec, rely 255/255, load 1/255
```

这是在 Catalyst 交换机环境中的一个常见设置。当在一个交换机的 FDDI 端口收到一个分组的时候，它穿过交换机背板到达安装了 RSM 的插槽。从 FDDI 到以太网的转换 / 分段过程在交换机一级发生。

有着 MTU 不匹配发现 (MUT mismatch-detection) 特性，这两个路由器将不会形成一个邻接体关系。在这个特定情况下，接口级命令 **ip ospf mtu-ignore** 被加入到了 Cisco IOS 软件版本 12.1.3 和更高版本中。这个命令忽视 FDDI MTU 并在这种特定情况下形成一个邻接体关系。这个命令不应该在任何其它情况下使用，因为 MTU 不匹配检测对疑难解析很重要。要使用这个命令，需在接口上应用它。在这个例子中，应该在 VLAN 1 接口上应用它。

例 9-91 显示了在解决这个 MTU 问题后 **show ip ospf neighbor** 的输出。

例 9-91 确认 MTU 不匹配问题已经解决了

```
R2#show ip ospf neighbor
Neighbor ID    Pri  State           Dead Time   Address      Interface
131.108.2.1    1    FULL/DR         00:00:32   131.108.1.1  Fddi0
```

2. OSPF 邻居停滞于 EXSTART / EXCHANGE 状态——原因：在邻居上有重复的路由器 ID

当 OSPF 发送一个 DBD 分组来选举一个主设备和从设备的时候，具有最大路由器 ID 的路由器变成主设备。这发生在 EXSTART 过程中。如果选举中有任何问题，路由器将停滞于 EXSTART / EXCHANGE 状态。

图 9-35 显示了解决这个问题应遵循的流程图。

(1) 调试与验证

例 9-92 显示了 R1 上 **show ip ospf neighbor** 的输出，表明邻居停滞于 EXSTART 状态。

例 9-92 **show ip ospf neighbor** 命令输出显示 R1 处于 EXSTART 状态

```
R2#show ip ospf neighbor
Neighbor ID    Pri  State           Dead Time   Address      Interface
131.108.2.1    1    EXSTART/-       00:00:33   131.108.1.1  Serial0
```

例 9-93

这就表明有一从设备。一个标志值在第一的更多信息，

例 9-93

```
R2#debug
OSPF: R
OSPF: S
OSPF: R
state
EXSTART
OSPF: P
```

例 9-94 显

131.108.2.1—

例 9-94

```
R2#show
Serial0
Interne
Process
```

(2) 解决

例 9-93 显

到一个 DBD 分就发送 DBD 分停滞于 EXSTA

要解决这个如果是这样，你

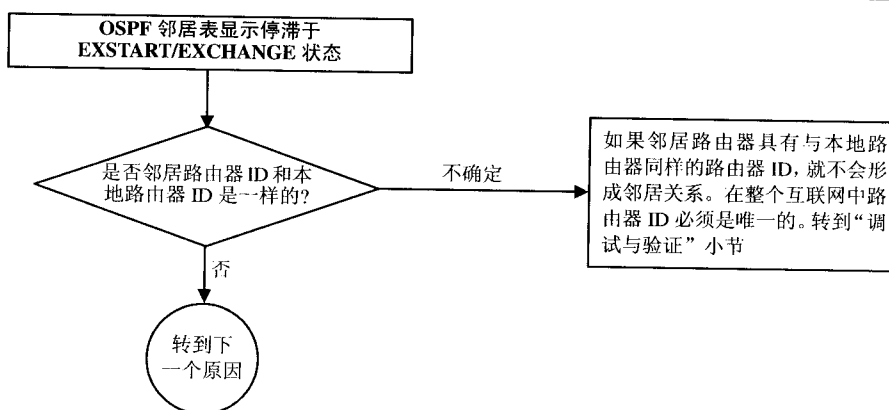


图 9-35 解决问题流程图

例 9-93 显示了 `debug ip ospf adj` 的输出。如果 DBD 分组持续重传并且标志值保持为 7，这就表明有一个问题。这表示没有路由器可以决定哪一个将成为主设备和哪一个将成为一个从设备。一个标志是从 DBD 分组格式中的一个 3 比特的值，它表示为 I、M 和 MS 比特。标志值在第一个 DBD 中设为 7，这表示 I、M 和 MS 比特都设置为 1。关于 I、M 和 MS 比特的更多信息，请参考第 8 章。

例 9-93 debug 输出显示一个主设备和从设备没有被形成

```

R2#debug ip ospf adj
OSPF: Retransmitting DBD to 131.108.2.1 on Serial0
OSPF: Send DBD to 131.108.2.1 on Serial0 seq 0x793 opt 0x2 flag 0x7 len 32
OSPF: Rcv DBD from 131.108.2.1 on Serial0 seq 0x25F7 opt 0x2 flag 0x7 len 32 mtu 0
state
EXSTART
OSPF: First DBD and we are not SLAVE
  
```

例 9-94 显示了在 R2 上 `show ip ospf interface serial0` 的输出，其中显示路由器 ID 为 131.108.2.1——和邻居的一样。这阻止了主从设备的选举。

例 9-94 R2 的路由器 IP 和邻居 R1 的一样

```

R2#show ip ospf interface serial 0
Serial0 is up, line protocol is up
Internet Address 131.108.1.2/24, Area 1
Process ID 1, Router ID 131.108.2.1, Network Type POINT_TO_POINT, Cost: 64
  
```

(2) 解决方法

例 9-93 显示了 R2 发送一个标志为 7 的 DBD 分组，说，“我是主设备”。R2 也从 R1 收到一个 DBD 分组说，“我是主设备”。R2 比较了 R1 的路由器 ID 发现它不比自己的高，所以就发送 DBD 分组到 R1 说，“我是主设备”。因此，两个路由器持续为主设备状态争斗因而都停滞于 EXSTART 状态。

要解决这个问题，需小心地检查邻居路由器 ID 和本地路由器 ID 看它们是否是相同的，如果是这样，你必须在其中一个路由器上改变路由器 ID 并重启 OSPF 协议从而使它生效。

注意: Cisco IOS 软件版本 12.0 和更高版本提供一个警告消息, OSPF-3-DUP_RTRID, 如果有重复的路由器 ID 它就报警。

例 9-95 显示了解决这个问题后 `show ip ospf neighbor` 的输出。

例 9-95 确认重复路由器 ID 的问题已经解决并且 OSPF 邻接体可以建立

R2#show ip ospf neighbor					
Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/-	00:00:32	131.108.1.1	Serial0

3. OSPF 邻居停滞于 EXSTART/EXCHANGE 状态——原因: 不能够用超过一定大小的 MTU ping 通

当 OSPF 开始与它的邻居形成一个邻接体关系的时候, 它要经过几个状态。在 EXSTART 状态中, OSPF 决定哪个路由器将成为主设备和哪个将成为从设备。在路由器做出了决定之后, 它们将开始以 DBD 分组的形式交换 LSA 头。如果数据库很大, OSPF 就使用接口 MTU 并且试图发送尽可能多的数据直至达到接口 MTU 的限制。如果存在一个第 2 层接收大小在接口 MTU 范围之内的大分组的问题, OSPF 邻接体关系将停滞于 EXCHANGE 状态。

图 9-36 显示了产生这个问题的网络设置。第 2 层介质有意地没有显示在这个图中, 因为这个问题可能发生在任何第 2 层介质中。



图 9-36 产生 MTU 问题的网络设置

例 9-96 显示了在 R2 上 `show ip ospf neighbor` 的输出, R2 与 R1 在串行链路上停滞于 EXCHANGE 状态。这表示主从设备协商已经发生了。

例 9-96 `show ip ospf neighbor` 命令输出表示一个 EXSTART 问题

R2#show ip ospf neighbor					
Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	EXCHANGE/-	00:00:46	131.108.1.2	Serial0/0

图 9-37 显示了解决这个问题应遵循的流程图。

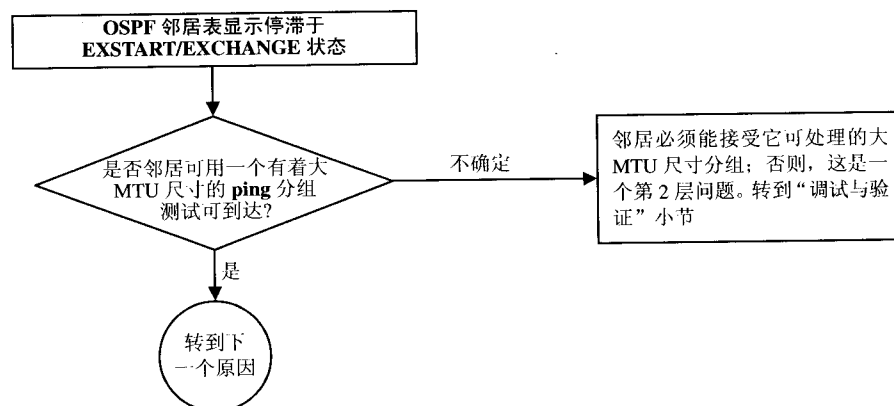


图 9-37 解决问题流程图

(1) 调试与验证

例 9-97 显示了 **debug ip ospf adj** 的输出。调试显示 R2 持续每间隔 5 秒钟重传一次 DBD 分组, 5 秒钟时间是缺省值, 并且没有收到任何回应。也注意到分组长度为 1274, 标志值为 3; 这表示 R2 是主设备。回忆前面的问题, 标志值是 3 表示 M 和 MS 比特被设置了。

例 9-97 debug ip ospf adj 命令输出显示 DBD 分组传输历史

```
R2#debug ip ospf adj
OSPF: Send DBD to 131.108.2.1 on Serial0 seq 0x793 opt 0x2 flag 0x3 len 1274
OSPF: Retransmitting DBD to 131.108.2.1 on Serial0
OSPF: Send DBD to 131.108.2.1 on Serial0 seq 0x793 opt 0x2 flag 0x3 len 1274
OSPF: Retransmitting DBD to 131.108.2.1 on Serial0
OSPF: Send DBD to 131.108.2.1 on Serial0 seq 0x793 opt 0x2 flag 0x3 len 1274
OSPF: Retransmitting DBD to 131.108.2.1 on Serial0
OSPF: Send DBD to 131.108.2.1 on Serial0 seq 0x793 opt 0x2 flag 0x3 len 1274
OSPF: Retransmitting DBD to 131.108.2.1 on Serial0
```

例 9-98 显示了从 R1 到 R2 的普通和扩展 ping 的输出。当 R1 用有一个等于或超过 1200 的 MTU ping R2 时, ping 将不会到达另一边。这表明在第 2 层有一个问题。

例 9-98 普通 ping 成功了但有着 1200 MTU 的 ping 失败了

```
R1#ping 131.108.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 131.108.1.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R1#

R1#ping ip
Target IP address: 131.108.1.2
Repeat count [5]:
Datagram size [100]: 1200
Timeout in seconds [2]:
Extended commands [n]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 1200-byte ICMP Echos to 131.108.1.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R1#
```

(2) 解决方法

这个问题实际上与第 2 层有关。当使用一个 100 字节数据的时候 R1 可以 ping 通 R2, 但是当数据报尺寸大于 1200 字节的时候 ping 开始失败了。

要解决这个问题, 首先应该解决第 2 层的问题。一个缩小范围的方法是将这两个设备直接相连而不是通过交换机相连, 看是否这个问题是出在第 2 层设备或者路由器本身上。如果背对背连接路由器不能解决问题, 就存在硬件出错的可能。大多数时候, 证明是中间出问题

了——例如，一个 LAN 交换机或者一个电信云出了问题。

依据介质不同，有几个建议：

- 在一个 LAN 介质的情况中：
 - 检查在这种介质的交换机配置中定义的 MTU 尺寸。
 - 尝试使用一个不同的端口。
- 在一个 WAN 介质的情况中：
 - 如果你是 WAN 云的提供商，检查它在哪一跳失效。
 - 如果你从电信得到一条线路，请求检查中间的 WAN 云来发现它在哪里出错。

4. OSPF 邻居停滞于 EXSTART/EXCHANGE 状态——原因：单播连接中断

当 OSPF 路由器开始相互交换数据库信息的时候，它们在 EXSTART / EXCHANGE 状态相互发送一个单播分组。这仅仅在网络类型不是一个点到点链路的时候发生。在点到点链路的情况中，OSPF 全部发送组播分组。如果单播连接中断了，OSPF 邻居将停留在 EXSTART 状态。

图 9-38 显示了解决这个问题应遵循的流程图。

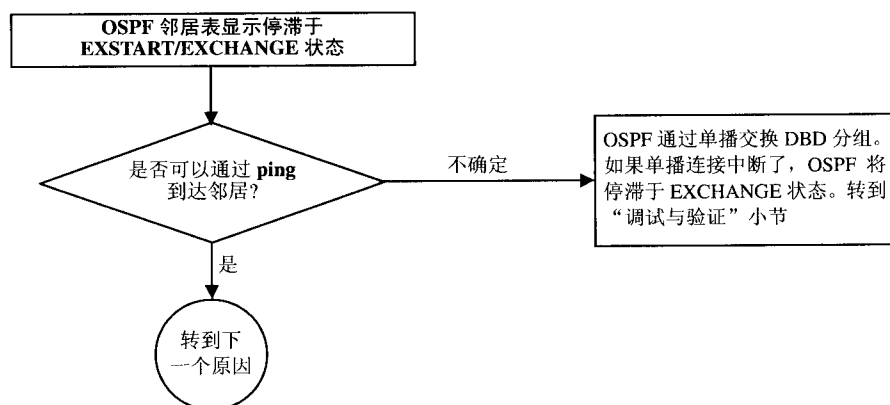


图 9-38 解决问题流程图

(1) 调试与验证

例 9-99 显示了从 R1 到 R2 的一个 ping 的输出，输出显示一个有着 100 字节数据报的 ping 分组丢失了。

例 9-99 ping 失败显示单播连接中断了

```

R1#ping 131.108.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 131.108.1.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R1#
  
```

(2) 解决方法

ping 失败可能有几个原因，包括如下：

- 在帧
- 一个
- NAT

错误的

在帧中继

丢失。为了进

命令。

例 9-100

了帧中继云，

例 9-100

```

R1#show
Extende
perm
R1#debu
R1#ping
  
```

```

Type es
Sending
.....
Success
  
```

```

IP: s=1
ICM
IP: s=1
ICM
IP: s=1
ICM
IP: s=1
ICM
IP: s=1
ICM
IP: s=1
ICM
R1#
  
```

```

R2#show
Extende
perm
R2#debu
R2#ping
Type es
Sending
.....
Success
  
```

```

IP: s=1
ICM
  
```

- 在帧中继交换机或者 ATM 交换机中存在一个错误的 DLCI 或者 VPI / VCI 映射。
- 一个访问列表阻止单播。
- NAT 对单播进行了转换。

错误的 DLCI 或者 VPI / VCI 映射

在帧中继或者 ATM 的情形中，这是一个很常见的错误。分组将在帧中继或者 ATM 云中丢失。为了进一步确认是这种情况，在两边路由器上都对访问列表打开 **debug ip packet detail** 命令。

例 9-100 显示在 R1 和 R2 上 **debug ip packet detail** 的输出，表示 ICMP 分组已被发送进了帧中继云，但是没有东西返回来。

例 9-100 debug ip packet detail 命令表示 ICMP 分组传输成功但是没有接收到任何应答

```
R1#show access-list 100
Extended IP access list 100
  permit ip 131.108.1.0 0.0.0.255 131.108.1.0 0.0.0.255 (10 matches)
R1#debug ip packet detail 100
R1#ping 131.108.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 131.108.1.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0), len 100, sending
  ICMP type=8, code=0
IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0), len 100, sending
  ICMP type=8, code=0
IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0), len 100, sending
  ICMP type=8, code=0
IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0), len 100, sending
  ICMP type=8, code=0
IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0), len 100, sending
  ICMP type=8, code=0
R1#
```

```
R2#show access-list 100
Extended IP access list 100
  permit ip 131.108.1.0 0.0.0.255 131.108.1.0 0.0.0.255 (10 matches)
R2#debug ip packet detail 100
R2#ping 131.108.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 131.108.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

IP: s=131.108.1.2 (local), d=131.108.1.1 (Serial0), len 100, sending
  ICMP type=8, code=0
```

(待续)

```

IP: s=131.108.1.2 (local), d=131.108.1.1 (Serial0), len 100, sending
    ICMP type=8, code=0
IP: s=131.108.1.2 (local), d=131.108.1.1 (Serial0), len 100, sending
    ICMP type=8, code=0
IP: s=131.108.1.2 (local), d=131.108.1.1 (Serial0), len 100, sending
    ICMP type=8, code=0
IP: s=131.108.1.2 (local), d=131.108.1.1 (Serial0), len 100, sending
    ICMP type=8, code=0
R2#

```

要解决这个问题，应该联系电话运营商以决定是否发生了这样的事。这个问题与路由器本身有关从而导致路由器丢失分组的这样一种可能性是很小的。任何其它的问题将出现在调试消息中。例如在路由器中错误的帧中继续映射这样的问题导致了在调试输出中的“封装失败”消息。

访问列表阻止单播

如果在路由器上配置了一个访问列表，请确信它不阻止单播分组。例 9-101 显示了在 R2 上 **debug ip packet detail 100** 的输出，它显示单播被阻止了。访问列表 101 显示只有 OSPF 的组播分组被允许，而从地址 131.108.1.0 来的单播分组被拒绝了，因为有一个明确的 **deny** 位于每个访问列表的末尾。

例 9-101 揭示单播连接被阻止了

```

R1#show access-list 100
Extended IP access list 100
    permit ip 131.108.1.0 0.0.0.255 131.108.1.0 0.0.0.255
R1#show access-list 101
Extended IP access list 101
    permit ip 141.108.10.0 0.0.0.255 any
    permit ip 141.108.20.0 0.0.0.255 any
    permit ip 141.108.30.0 0.0.0.255 any
    permit ip 131.108.1.0 0.0.0.255 host 224.0.0.5
R1#debug ip packet 100 detail
IP packet debugging is on (detailed) for access list 100
R1#
IP: s=131.108.1.2 (Serial0.2), d=131.108.1.1, len 100, access denied
    ICMP type=8, code=0
IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0.2), len 56, sending
    ICMP type=3, code=13
R1#
IP: s=131.108.1.2 (Serial0.2), d=131.108.1.1, len 100, access denied
    ICMP type=8, code=0
IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0.2), len 56, sending
    ICMP type=3, code=13
R1#
IP: s=131.108.1.2 (Serial0.2), d=131.108.1.1, len 100, access denied
    ICMP type=8, code=0
IP: s=131.108.1.1 (local), d=131.108.1.2 (Serial0.2), len 56, sending
    ICMP type=3, code=13
R1#

```

例 9-101

确的 **deny** 在播分组)。这导致要解决这个问题，访问列表以解

例 9-102

```

R1#show
Extended
    permit
    permit
    permit
    permit
    permit

```

对单播作

这是在路由转换冲它而来的口是 Serial0.2,

10.

当 R2 发送应。要观察的主例 9-98 显示了在

例 9-103

```

R1#
interface
ip nat out
!
ip nat ins
!
access-lis

```

要解决这个问题，了解这个问题，思就是访问列表 10.0.0.0 / 8 被允

例 9-101 清楚地显示了分组被拒绝是因为访问列表的原因。所有的访问列表都有一个明确的 **deny** 在表的末尾。因此它们也拒绝任何没有被明确允许的分组（在这个情况中，是单播分组）。这导致 OSPF 停滞于 EXCHANGE 状态。

要解决这个问题，需修改访问列表 101，让它允许单播分组。例 9-102 显示了修改后的访问列表以解决这个问题。

例 9-102 修改一个访问列表来允许单播分组

```
R1#show access-list 101
Extended IP access list 101
  permit ip 141.108.10.0 0.0.0.255 any
  permit ip 141.108.20.0 0.0.0.255 any
  permit ip 141.108.30.0 0.0.0.255 any
  permit ip 131.108.1.0 0.0.0.255 host 224.0.0.5
  permit ip 131.108.1.0 0.0.0.255 131.108.1.0 0.0.0.255
```

对单播作了 NAT 转换

这是在路由器上配置了 NAT 时发生的另一个常见问题。如果 NAT 配置错了，它将开始转换冲它而来的单播分组，这将中断单播连接。例 9-103 显示 R1 配置了 NAT。R1 的外部接口是 Serial0.2，它连接 R2。图 9-39 显示了 R1 和 R2 互连，而 R1 运行 NAT。

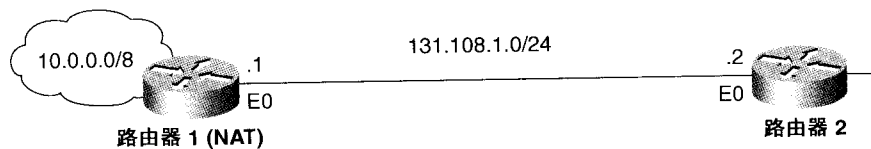


图 9-39 R1 运行 NAT 的网络

当 R2 发送单播分组给 R1 时，R1 试图转换接收到的分组，因而 R2 不会收到 ping 的回应。要观察的主要事情是 NAT 中的访问列表。如果访问列表允许任何事情，这个问题将发生。例 9-98 显示了在 R1 上的 NAT 配置。

例 9-103 NAT 配置导致单播分组被转换

```
R1#
interface Ethernet 0
ip nat outside
!
ip nat inside source list 1 interface Serial0.2 overload
!
access-list 1 permit any
```

要解决这个问题，可改变 access-list 1 仅仅允许这些需要转换的 IP 地址，例 9-104 显示了解决这个问题的正确的访问列表。从一个网络到另一个网络访问列表可能不同。整个的意思就是访问列表的 **permit** 语句不应该覆盖邻居的 IP 地址。在例 9-104 中，仅仅内部网络 10.0.0.0 / 8 被允许。这表示 R1 将不再转换属于 131.108.1.0 网络的分组。

例 9-104 修正访问列表来解决单播连接问题

```

R1#
interface Ethernet 0
ip address 131.108.1.1 255.255.255.0
ip nat outside
!
ip nat inside source list 1 interface Serial0.2 overload
!
access-list 1 permit 10.0.0.0 0.255.255.255

```

例 9-105 显示了 **show ip ospf neighbor** 的输出, 其中显示了在解决这个单播问题后, OSPF 邻居处于 FULL 状态。

例 9-105 确认单播问题已经解决了

```

R2#show ip ospf neighbor
Neighbor ID      Pri  State      Dead Time   Address      Interface
131.108.2.1      1    FULL/-     00:00:32    131.108.1.1  Ethernet0

```

5. OSPF 邻居停滞于 EXSTART / EXCHANGE 状态——原因: 在 PRI 和 BRI / 拨号接口之间网络类型为点到点类型

在一个 PRI 接口网络类型是点到点类型。这引起 OSPF 甚至在 2-way 状态之后还发送组播分组。如果仅有一个 BRI 作为一个 OSPF 邻居, 它将工作得很好。可是, 当多个 BRI 尝试与这个 PRI 形成一个邻接体时, PRI 将报错, 因为它的网络类型是点到点的。因为所有 OSPF 分组被作为组播分组在一个点到点链路上发送, PRI 从多个 BRI 邻居接收 DBD 分组, 这就会导致所有邻居进入 EXSTART / EXCHANGE 状态。

图 9-40 显示了产生这个问题的网络设置。R1 有一个 PRI, R2 和 R3 拨号到这个 PRI 中。这在 OSPF 中产生了一个问题, 因为网络类型是点到点的。

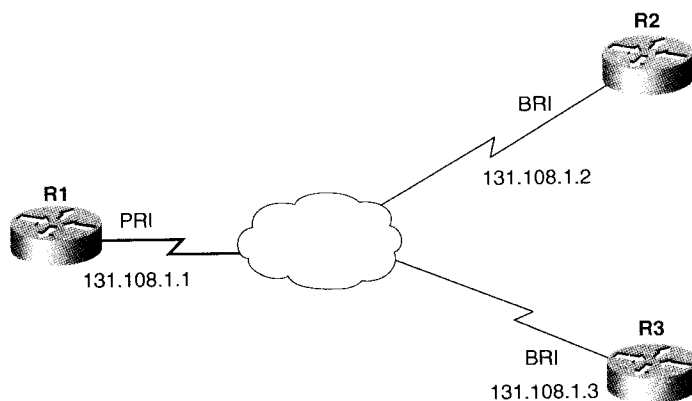


图 9-40 网络类型为点到点类型的 PRI 到 BRI 设置

例 9-106 显示了在 R1 上 **show ip ospf neighbor** 的输出。R2 和 R3 都停滞于 EXSTART 状态, R1 处于一个 ISDN 链路上。如果输出显示邻居处于 EXSTART 状态中持续很长一段时间,

这将是一个问题的征兆。

例 9-106 PRI 邻居停滞于 EXSTART 状态

```
R1#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.1.2	1	EXSTART/-	00:00:38	131.108.1.2	Serial0/0:23
131.108.1.3	1	EXSTART/-	00:00:32	131.108.1.3	Serial0/0:23

图 9-41 显示了解决这个问题应遵循的流程图。

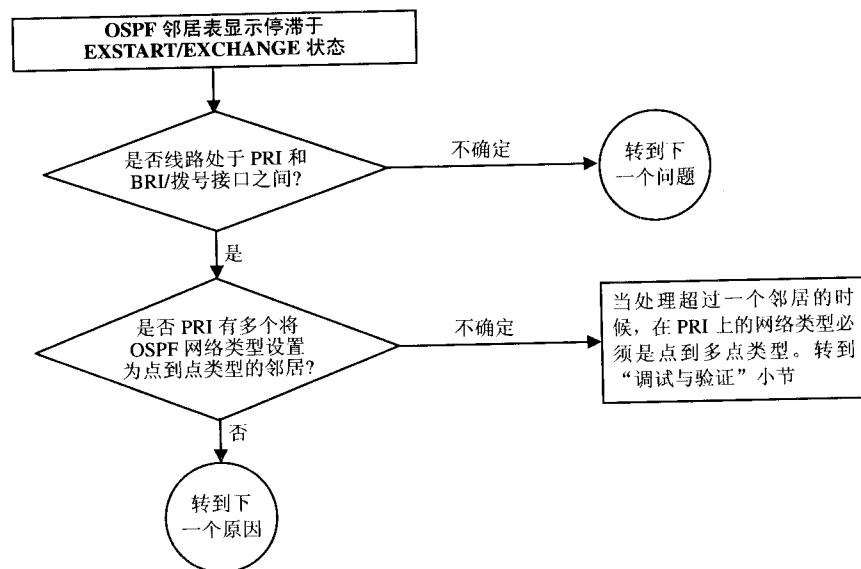


图 9-41 解决问题流程图

(1) 调试与验证

例 9-107 显示在 R2 上 `show ip ospf interface bri0` 的输出，表示网络类型为点到点类型。

例 9-107 确认在 R2 的 bri0 接口上的网络类型

```
R2#show ip ospf interface bri0
BRI0 is up, line protocol is up (spoofing)
Internet Address 131.108.1.2/24, Area 2
Process ID 1, Router ID 131.108.1.2, Network Type POINT_TO_POINT, Cost: 1562
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:06
index 1/1, flood queue length 0
Next 0x0(0)/0x0(0)
Last flood scan length is 1, maximum is 1
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 1, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)
```

例 9-108 显示了在 R2 上 `debug ip ospf adj` 的输出。调试显示 R2 在一个点到点网络类型

上接收了两个不同的 DBD 分组。问题是当 R1 发送 DBD 分组给 R2 和 R3 时，它将它们作为组播分组发送，因为网络类型被定义为点到点类型。在点到点网络中，所有 OSPF 分组都被作为组播分组发送。这导致 R2 接收目的地址为 R3 的 DBD 分组，反之亦然。

当 R2 收到一个 DBD 分组的时候，它会报错，因为这个 DBD 分组的序列号和标志是不同的。这导致 R2 回退到 EXSTART 状态。这个循环持续重复。

例 9-108 debug 输出显示 R2 收到了 R3 的 DBD 分组，这引起了问题

```
R2#debug ip ospf adj
Send DBD to 131.108.1.1 on BRI0 seq 0xB41 opt 0x42 flag 0x7 len 32
Rcv DBD from 131.108.1.1 on BRI0 seq 0x1D06 opt 0x42 flag 0x7 len 32 mtu 1500 state
EXSTART
First DBD and we are not SLAVE
Rcv DBD from 131.108.1.1 on BRI0 seq 0xB41 opt 0x42 flag 0x2 len 92 mtu 1500 state
EXSTART
NBR Negotiation Done. We are the MASTER
Send DBD to 131.108.1.1 on BRI0 seq 0xB42 opt 0x42 flag 0x3 len 92
Database request to 131.108.1.1
sent LS REQ packet to 131.108.1.1, length 12
Rcv DBD from 131.108.1.1 on BRI0 seq 0x250 opt 0x42 flag 0x7 len 32 mtu 1500 state
EXCHANGE
EXCHANGE - inconsistent in MASTER/SLAVE
Bad seq received from 131.108.1.1 on BRI0
Send DBD to 131.108.1.1 on BRI0 seq 0x2441 opt 0x42 flag 0x7 len 32
Rcv DBD from 131.108.1.1 on BRI0 seq 0x152C opt 0x42 flag 0x2 len 92 mtu 1500 state
EXSTART
Unrecognized dbd for EXSTART
Rcv DBD from 131.108.1.1 on BRI0 seq 0xB42 opt 0x42 flag 0x0 len 32 mtu 1500 state
EXSTART
Unrecognized dbd for EXSTART
```

(2) 解决方法

要解决这个问题，需改变 PRI 和 BRI 接口的网络类型为点到多点类型。例 9-109 显示了改变网络类型为点到多点类型的接口级命令，后面跟着在 R2 上 **show ip ospf interface** 的输出。

例 9-109 确认在 R2 的 bri 0 接口上的网络类型

```
R2#
interface BRI0
 ip ospf network point-to-multipoint

R2#show ip ospf interface bri0
BRI0 is up, line protocol is up (spoofing)
Internet Address 131.108.1.2/24, Area 2
Process ID 1, Router ID 131.108.1.2, Network Type POINT_TO_MULTIPOINT, Cost: 1562
Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT,
Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
Hello due in 00:00:06
index 1/1, flood queue length 0
```

(待续)

这个改
使得 OSPF
地址为 R3

9.2.

这是一
由器已经向
的邻居来的
路由器将停

这个问题

● 不

● 损

图 9-4

个问题。

131.108

例 9-1

态。

例 9-1

R2#sh

Neigh

131.

1. O

这是
这个问题就
的 MTU
LOADING

图 9-4

```

Next 0x0(0)/0x0(0)
Last flood scan length is 1, maximum is 1
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 1, Adjacent neighbor count is 1
Suppress hello for 0 neighbor(s)

```

这个改变必须在连接到 ISDN 云上的所有路由器上进行。改变网络类型为点到多点类型使得 OSPF 在 2-way 状态之后为 DBD 分组发送一个单播分组而不是一个组播分组,因而目的地址为 R3 的分组不会到达 R2。

9.2.6 问题: OSPF 邻居停滞于 LOADING 状态

这是一个 OSPF 邻居关系中少见的问题。当一个邻居停滞于 LOADING 状态时,本地路由器已经向邻居发送了一个链路状态请求分组去请求一个过期或丢失的 LSA,并且等待从它的邻居来的一个路由更新分组。如果邻居没有回答或邻居的回答没有到达本地路由器,那么路由器将停滞于 LOADING 状态。

这个问题最常见的可能原因如下:

- 不匹配的 MTU。
- 损坏的链路状态请求分组。

图 9-42 显示了有两个运行 OSPF 的路由器的网络, R1 正遭受停滞于 LOADING 状态这个问题。



图 9-42 OSPF 邻居停滞于 LOADING 状态问题的网络拓扑

例 9-110 显示了 `show ip ospf neighbor` 的输出,表示 R2 的邻居正停滞于 LOADING 状态。

例 9-110 `show ip ospf neighbor` 命令输出表明邻居的状态——这里为 LOADING

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	LOADING/-	00:00:37	131.108.1.1	Serial0

1. OSPF 停滞于 LOADING 状态——原因: 不匹配的 MTU 大小

这是当一个 MTU 不匹配产生时发生的一个特殊问题。如果在链路上 MTU 大小不一致,这个问题就会发生。特别地,如果一个邻居的 MTU 比本地路由器的要大,邻居就会发送大的 MTU 分组作为一个链路状态更新。这个分组不会到达本地路由器,因此邻居停滞于 LOADING 状态。

图 9-42 显示了解决这个问题应遵循的流程图。

(待续)

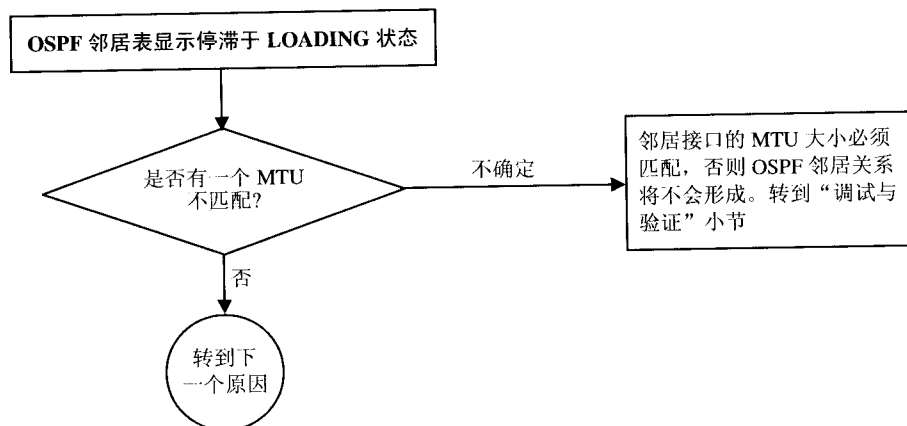


图 9-43 解决问题流程图

(1) 调试与验证

例 9-111 显示了在 R1 和 R2 上的接口配置。两个配置都显示 MTU 值互不相同。

例 9-111 R1 和 R2 的配置有不同的 MTU 值

```

R2#show interface Serial0
Serial0/0 is up, line protocol is up
Hardware is PQUICC with Fractional T1 CSU/DSU
MTU 2048 bytes, BW 256 Kbit, DLY 20000 usec,

R1#show interface ATM4/0/0
ATM4/0/0 is up, line protocol is up
Hardware is cyBus ATM
MTU 4470 bytes, sub MTU 4470, BW 155520 Kbit, DLY 80 usec,
  
```

例 9-112 显示了 R1 和 R2 运行的 Cisco IOS 软件版本，因为 R2 运行 11.3 (10) T，该版本比 12.0.3 低，不能检测不匹配的 MTU。MTU 不匹配检测被加入到了 RFC 2178 并在 Cisco IOS 软件版本 12.0.3 和更高版本中实现。

例 9-112 确认在 R1 和 R2 上使用的 Cisco IOS 软件版本

```

R2#show version
Cisco Internetwork Operating System Software
IOS (tm) C2600 Software (C2600-I-M), Version 11.3(10)T, RELEASE SOFTWARE (fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.

R1#show version
Cisco Internetwork Operating System Software
IOS (tm) RSP Software (RSP-JSV-M), Version 12.0(7)T, RELEASE SOFTWARE (fc2)
Copyright (c) 1986-1999 by cisco Systems, Inc.
  
```

例 9-113 显示了在 R2 上 `debug ip ospf adj` 的输出。调试显示 R2 连续地向 R1 重传 DBD 分组，但是 R1 的回应不会到达 R2，因为分组太大了。

例 9-11

```

R2#deb
OSPF a
R2#
OSPF:
OSPF:
OSPF:
OSPF:
  
```

(2) 解

在这个配检测。R1 的 MTU 比配检测功能在这里代码，R1 也

当 R2 R2 不会得到个接口（在

```

inter
mtu 4
  
```

例 9-11

于 FULL 状

例 9-1

```

R2#sh
Neigh
131.1
  
```

2. OS

当一个应，这导致链路状

- 一
- 发
- 是
- 接
- 场

图 9-4

(1) 讲

例 9-1 组。这是分

例 9-113 在 R2 上 debug ip ospf adj 命令输出显示到 R1 的 DBD 分组的传输历史

```

R2#debug ip ospf adj
OSPF adjacency events debugging is on
R2#
OSPF: Retransmitting request to 131.108.2.1 on Serial0
OSPF: Database request to 131.108.2.1
OSPF: sent LS REQ packet to 131.108.1.1, length 12
OSPF: Retransmitting request to 131.108.2.1 on Serial0

```

(2) 解决方法

在这个特定的情况中, R2 运行 Cisco IOS 软件版本 11.3.10T, 该版本不支持 MTU 不匹配检测。R1 运行 Cisco IOS 软件版本 12.0.7T, 该版本支持 MTU 不匹配检测。R1 仅仅当 R2 的 MTU 比 R1 的大时才检测 MTU 不匹配问题; 否则, 它不会报错。换句话说, MTU 不匹配检测功能仅仅是在存在一个有着比本地路由器更大 MTU 的邻居的时候才生效。

在这里, R2 的 MTU 是 2048, 因此尽管 R1 运行有着 MTU 不匹配检测的 Cisco IOS 软件代码, R1 也不能检测到一个 MTU 不匹配, 因为 R2 的 MTU 比 R1 的小。

当 R2 为新的 LSA 实例发送 LS 请求分组时, R1 回应一个超过 2048 字节的 LSA, 因此 R2 不会得到那个分组, 因为它太大了。要解决这个问题, 请确信两边的 MTU 匹配。要在一个接口(在这里, 是 R2 的 Serial0 接口)改变 MTU 大小, 请键入下面的接口级命令:

```

interface serial 0
mtu 4470

```

例 9-114 显示了 show ip ospf neighbor 的输出, 表明在解决单播问题之后 OSPF 邻居处于 FULL 状态。

例 9-114 确认在解决 MTU 问题后 OSPF 形成了邻居关系

```

R2#show ip ospf neighbor
Neighbor ID    Pri   State       Dead Time   Address        Interface
131.108.2.1    1     FULL/-      00:00:32    131.108.1.1    Serial0

```

2. OSPF 邻居停滞于 LOADING 状态——原因: 链路状态请求分组损坏了

当一个链路状态请求分组损坏了, 邻居丢弃这个分组而本地路由器不会从邻居收到回应, 这导致 OSPF 邻居停滞于 LOADING 状态。

链路状态请求分组通常因为以下原因而损坏:

- 一个在邻居中间的设备(比如交换机)破坏了这个分组。
- 发送方路由器的分组是无效的。在这种情况下, 不是发送方路由器的接口损坏了就是由软件 bug 导致了错误。
- 接收方路由器计算出了错误的校验和。在这种情况下, 不是接收方路由器的接口损坏了就是软件 bug 导致了错误。这种出错的可能性最小。

图 9-44 显示了解决这个问题应遵循的流程图。

(1) 调试与验证

例 9-115 显示了在 R2 上的日志消息, 表示 R2 接收到了一个有着错误校验和的 OSPF 分组。这是分组损坏的标志。

例 9-115 日志显示 OSPF 收到了损坏的分组

```
R2#show log
```

```
%OSPF-4-ERRRCV: Received invalid packet: Bad Checksum from 131.108.1.1, Serial0
%OSPF-4-ERRRCV: Received invalid packet: Bad Checksum from 131.108.1.1, Serial0
```

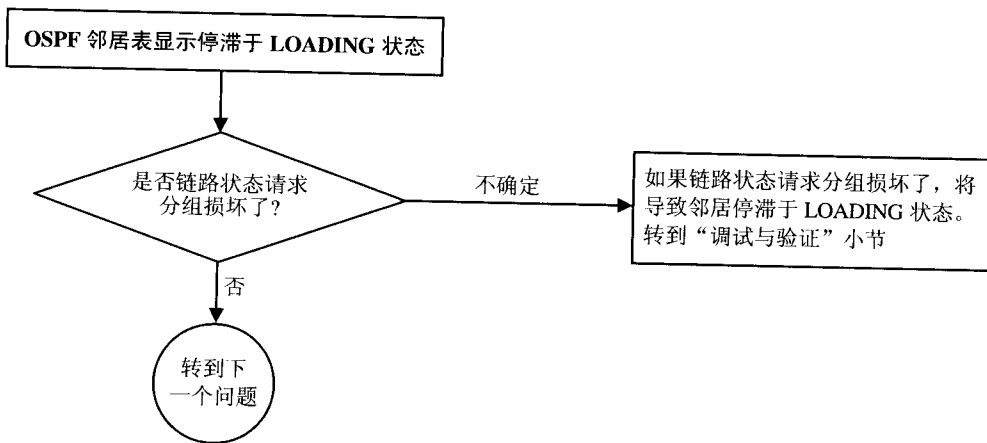


图 9-44 解决问题流程图

例 9-116 显示了 R2 重传 LS 请求分组并且没有得到任何应答, 因为应答被损坏了。

例 9-116 因为分组的损坏, R2 没有收到对它的链路状态请求分组的回应

```
R2#debug ip ospf adj
```

```
OSPF adjacency events debugging is on
```

```
R2#
```

```
OSPF: Retransmitting request to 131.108.2.1 on Serial0
```

```
OSPF: Database request to 131.108.2.1
```

```
OSPF: sent LS REQ packet to 131.108.1.1, length 12
```

```
OSPF: Retransmitting request to 131.108.2.1 on Serial0
```

(2) 解决方法

大多数的时候, 这个问题可以通过硬件替换来解决。这个问题可能是由于交换机上一个简单的损坏的端口或者发送 / 接收方路由器上一个损坏的接口卡而引起的。

例 9-117 显示了 **show ip ospf neighbor** 的输出, 表明 OSPF 邻居在解决了这个损坏的链路状态请求分组问题后处于 FULL 状态。

例 9-117 确认损坏的链路状态请求分组问题解决了, 允许形成一个 OSPF 邻接体关系

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/-	00:00:32	131.108.1.1	Serial0

本节讨论
系时, 它与
享, 那么 OS

OSPF 不

● OSPF

● OSPF

● OSPF

● OSPF

接下来这

9.3.1

当一个邻
个邻居没有包

这个问题

● 在假定

● 通告报

● 第二报

图 9-45 显

131.108.2.0/2

例 9-118 显

例 9-118

```
R2#show ip
```

```
% Network
```

```
R2#
```

1. OSPF 令

OSPF 仅仅

9.3 OSPF 路由通告疑难解析

本节讨论与 OSPF 路由通告相关的问题。OSPF 是一个链路状态协议，当它形成邻居关系时，它与它的邻居（们）交换整个链路状态数据库。如果有任何数据库信息没有和邻居共享，那么 OSPF 链路状态特性就会打破。

OSPF 不能共享关于一条特定链路的数据库信息的最常见原因如下：

- OSPF 邻居不通告路由。
- OSPF 邻居（ABR）不通告汇总路由。
- OSPF 邻居不通告外部路由。
- OSPF 邻居不通告缺省路由。

接下来这一节讲述了这些问题、每一个问题的可能原因，以及解决问题的办法。

9.3.1 问题：OSPF 邻居不通告路由

当一个邻居不通告路由的时候，那条路由将不显示在本地路由器的路由表中。这表示这个邻居没有包含这条路由到它的数据库中，否则，本地路由器一定已经收到它了。

这个问题最常见的可能原因如下：

- 在假定要通告的接口上没有启用 OSPF。
- 通告接口关闭了。
- 第二接口与主接口处于不同的区域中。

图 9-45 显示了一个用于产生这个问题的 OSPF 网络设置。

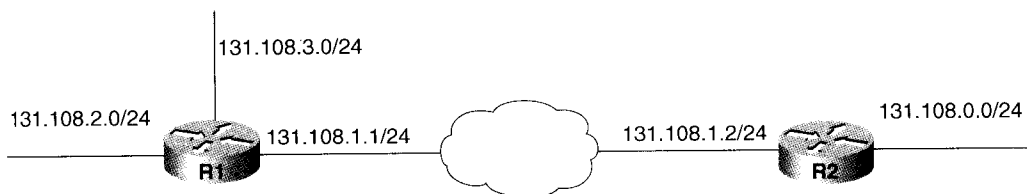


图 9-45 路由不能被成功通告的 OSPF 网络

例 9-118 显示了 **show ip route 131.108.3.0** 的输出，表明路由器 R2 的路由表中缺少这条路由。

例 9-118 R2 的路由表缺少路由 131.108.3.0

```
R2#show ip route 131.108.3.0
% Network not in table
R2#
```

1. OSPF 邻居不通告路由——原因：在假定要通告的接口上没有启用 OSPF

OSPF 仅仅当在那个接口上启用了 OSPF 的时候才在它的数据库中包含进那个接口的子

网地址。OSPF 可能在一个接口上不能启用是因为有一个不正确的 **network** 语句没有包含分配给一个接口的 IP 地址或者缺少这样一个 **network** 语句来包含这个接口 IP 地址。在这两种情况中，OSPF 都将从它的数据库中排除接口地址并将不通告给它的邻居。

图 9-46 显示了解决这个问题应遵循的流程图。

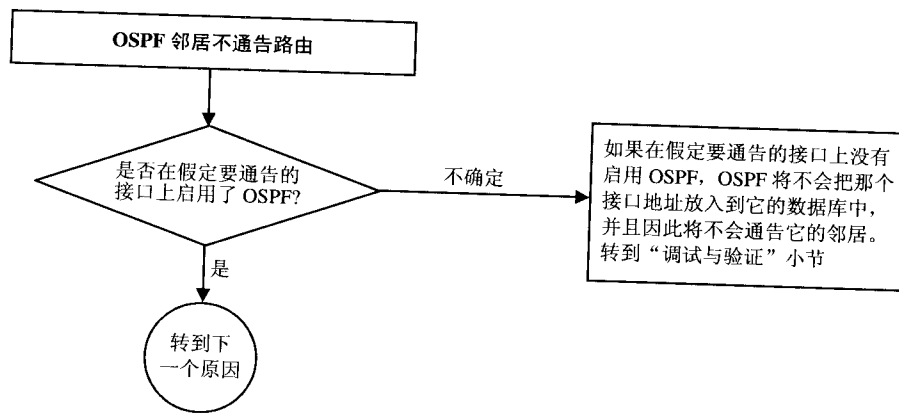


图 9-46 解决问题流程图

(1) 调试与验证

例 9-119 显示了在 R1 上 **show ip ospf database router** 的输出，表明数据库中缺少网络 131.108.3.0/24。

例 9-119 R1 的数据库缺少 131.108.3.0

```

R1#show ip ospf database router 131.108.2.1

OSPF Router with ID (131.108.1.2) (Process ID 1)

Router Link States (Area 0)
LS age: 301
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.2.1
Advertising Router: 131.108.2.1
LS Seq Number: 80000148
Checksum: 0x1672
Length: 48
Number of Links: 2

Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 131.108.1.2
(Link Data) Router Interface address: 131.108.1.1
Number of TOS metrics: 0
TOS 0 Metrics: 64

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.1.0
(Link Data) Network Mask: 255.255.255.0
  
```

(待续)

例 9-118 用。在 C 任何信息

例 9-119

R1#
Eth
OS

例 9-120

例 9-121

R1#
rout
net
net

(2) 角

R1 在

被通告。

问题是

语句。在这

LSA 的时候

要解决

路由器 LSA

例 9-122

例 9-123

例 9-124

R1#

router

netwo

netwo

netwo

netwo

例 9-125

在成为了一

语句没有包含分
地址。在这两种

上没有
会把那个
库中，
邻居。

数据库中缺少网络

```
Number of TOS metrics: 0
TOS 0 Metrics: 0
```

```
Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.2.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

例 9-120 显示了在 R1 上 **show ip ospf interface e0** 的输出, 表明 OSPF 在该接口上没有启用。在 Cisco IOS 软件版本 12.0 和更高版本中, 如果没有在该接口启用 OSPF, 则不会显示任何信息。

例 9-120 R1 在 Ethernet 0 接口上没有启用 OSPF

```
R1#show ip ospf interface Ethernet 0
Ethernet0 is up, line protocol is up
OSPF not enabled on this interface
```

例 9-121 显示了 R1 的配置, 它显示在配置中缺少对 131.108.3.0/24 的 **network** 语句。

例 9-121 R1 的配置缺少一个 131.108.3.0 的 network 语句

```
R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 0
network 131.108.2.0 0.0.0.255 area 0
```

(2) 解决方法

R1 在它的路由器 LSA 中没有通告它的 Ethernet 0 接口。数据库输出确认以太网子网没有被通告。

问题是在 R1 上用于以太网接口的 **network** 语句是不正确的或者在配置中缺少 **network** 语句。在这个情形中, OSPF 将不会包含特定的接口到路由器 LSA 中去; 当 R2 收到路由器 LSA 的时候, 这条特定链路没有包括进去。

要解决这个问题, 请确信 **network** 语句是正确的, 这样使得 R1 包含 131.108.3.0/24 到它的路由器 LSA 中。

例 9-122 显示了解决这个问题的正确配置。

例 9-122 纠正 R1 的配置使得在 131.108.3.0/24 上启用 OSPF

```
R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 0
network 131.108.2.0 0.0.0.255 area 0
network 131.108.3.0 0.0.0.255 area 0
```

例 9-123 显示了在解决这个问题之后 R1 的路由器 LSA。路由器 LSA 显示 131.108.3.0 现在成为了一条端链路。

(待续)

例 9-123 网络 131.108.3.0/24 现在处于 OSPF 数据库中

```

1#show ip ospf database router 131.108.2.1
    OSPF Router with ID (131.108.1.2) (Process ID 1)

    Router Link States (Area 0)

    LS age: 301
    Options: (No TOS-capability, DC)
    LS Type: Router Links
    Link State ID: 131.108.2.1
    Advertising Router: 131.108.2.1
    LS Seq Number: 80000148
    Checksum: 0x1672
    Length: 48
    Number of Links: 2

    Link connected to: another Router (point-to-point)
      (Link ID) Neighboring Router ID: 131.108.1.2
      (Link Data) Router Interface address: 131.108.1.1
      Number of TOS metrics: 0
      TOS 0 Metrics: 64

    Link connected to: a Stub Network
      (Link ID) Network/subnet number: 131.108.1.0
      (Link Data) Network Mask: 255.255.255.0
      Number of TOS metrics: 0
      TOS 0 Metrics: 0
R
    Link connected to: a Stub Network
      (Link ID) Network/subnet number: 131.108.2.0
      (Link Data) Network Mask: 255.255.255.0
      Number of TOS metrics: 0
      TOS 0 Metrics: 10

    Link connected to: a Stub Network
      (Link ID) Network/subnet number: 131.108.3.0
      (Link Data) Network Mask: 255.255.255.0
      Number of TOS metrics: 0
      TOS 0 Metrics: 10
  
```

例 9-124 显示了 **show ip route 131.108.3.0** 的输出，表明在解决了这个问题之后，路由在路由表中再次出现了。

例 9-124 确认路由 131.108.3.0 又可运作了

```

R2#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
  Known via "ospf 1", distance 110, metric 64, type intra area
  Redistributing via ospf 1
  
```

(待续)

2. 0
OSPF
器 LSA 中
图 9

(1) 记
回忆图
告到 OSPF
例 9-1

例 9-1

R1#sh
Ether
Inter
Proc
Trans
No d
No b
Timer

例 9-12

例 9-12

R2#shc

```

Last update from 131.108.1.1 on Serial0, 04:22:21 ago
Routing Descriptor Blocks:
* 131.108.1.1, from 131.108.2.1, 04:22:21 ago, via Serial0
Route metric is 64, traffic share count is 1

```

2. OSPF 邻居不通告路由——原因：通告接口关闭了

OSPF 不通告断开的网络。如果一个接口关闭了，分配给这个接口的网络将不会在路由器 LSA 中通告。

图 9-47 显示了解决这个问题应遵循的流程图。

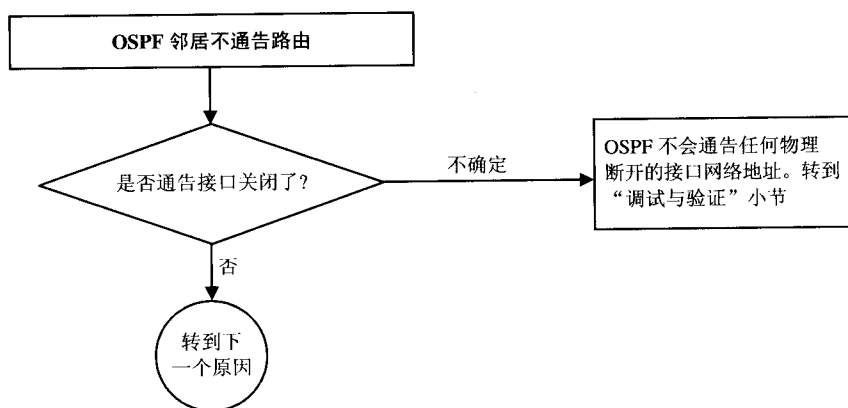


图 9-47 解决问题流程图

(1) 调试与验证

回忆图 9-45，Ethernet 0 接口地址为 131.108.3.0/24。如果这个接口关闭了，它将不被通告到 OSPF 数据库中。

例 9-125 显示对 Ethernet 0 接口的 **show ip ospf interface** 的输出，它表示线路协议关闭了。

例 9-125 **show ip ospf interface** 命令输出确定在 R1 上 Ethernet 0 接口的线路协议状态

```

R1#show ospf interface Ethernet 0
Ethernet0 is up, line protocol is down
Internet Address 131.108.3.1/24, Area 0
Process ID 1, Router ID 131.108.2.1, Network Type BROADCAST, Cost: 10
Transmit Delay is 1 sec, State DOWN, Priority 1
No designated router on this network
No backup designated router on this network
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5

```

例 9-126 显示了 Ethernet 0 接口的网络地址 131.108.3.0/24 没有列在 R1 的路由器 LSA 中。

例 9-126 R1 的 Ethernet 0 接口地址没有列在 R1 的 LSA 中

```

R2#show ip ospf database router 131.108.2.1

OSPF Router with ID (131.108.1.2) (Process ID 1)

```

(待续)

(待续)

Router Link States (Area 0)

```

LS age: 301
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.2.1
Advertising Router: 131.108.2.1
LS Seq Number: 80000148
Checksum: 0x1672
Length: 48
Number of Links: 2

Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 131.108.1.2
(Link Data) Router Interface address: 131.108.1.1
Number of TOS metrics: 0
TOS 0 Metrics: 64

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.1.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 0

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.2.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 10

```

(2) 解决方法

要解决这个问题, 需启用 Ethernet 0 接口。例 9-127 显示了解决了第 2 层问题后的接口统计数据。关于解决第 2 层问题的一些建议在第 9.3.2 节“OSPF 邻居表为空——原因: 第 1/2 层故障”中讨论过了。

例 9-127 确认为接口重新启用了线路协议

```

R1#show ip ospf interface Ethernet 0
Ethernet0 is up, line protocol is up
Internet Address 131.108.3.1/24, Area 0
Process ID 1, Router ID 131.108.2.1, Network Type BROADCAST, Cost: 10

```

例 9-128 显示了 **show ip route 131.108.3.0** 的输出, 其中显示了在解决这个问题后, 路由重新回到路由表中。

例 9-128 确认路由 131.108.3.0 可再次运作了

```

R2#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
Known via "ospf 1", distance 110, metric 64, type intra area
Redistributing via ospf 1

```

(待续)

```

Last update from 131.108.1.1 on Serial0, 04:22:21 ago
Routing Descriptor Blocks:
* 131.108.1.1, from 131.108.2.1, 04:22:21 ago, via Serial0
  Route metric is 64, traffic share count is 1

```

R2#

3. OSPF 邻居不通告路由——原因：第二接口与主接口处于不同的区域中

当处理第二地址时，OSPF 需要第二地址属于与主地址相同的区域。如果不是这样的，OSPF 将不在它的路由器 LSA 中通告第二地址。

图 9-48 显示了用于产生这个问题的网络设置。



图 9-48 用于产生第二地址 / LSA 问题的 OSPF 网络设置

图 9-49 显示了解决这个问题应遵循的流程图。

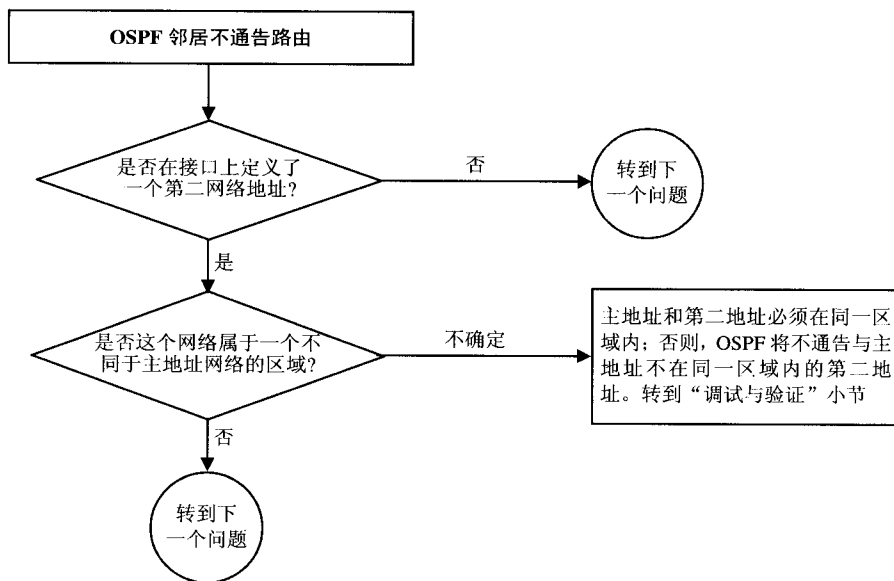


图 9-49 解决问题流程图

(1) 调试与验证

例 9-129 显示了 R1 的路由器 LSA 的信息。LSA 显示 131.108.3.0/24 没有被包括进数据库中。R1 为区域 2 产生路由器 LSA，但因为第二地址是在区域 2 中的唯一地址，链路编号为 0。这是因为第二接口地址不能被放进一个与主接口地址分离的区域中。

例 9-129 显示 R1 的路由器 LSA 信息

```
R1#show ip ospf database router 131.108.2.1
```

```
OSPF Router with ID (131.108.1.2) (Process ID 1)
```

(待续)

(待续)

Router Link States (Area 0)

```

LS age: 301
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.2.1
Advertising Router: 131.108.2.1
LS Seq Number: 80000148
Checksum: 0x1672
Length: 48
Number of Links: 2

Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 131.108.1.2
(Link Data) Router Interface address: 131.108.1.1
Number of TOS metrics: 0
TOS 0 Metrics: 64

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.1.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 0

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.2.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 10

```

Router Link States (Area 2)

```

LS age: 39
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.2.1
Advertising Router: 131.108.2.1
LS Seq Number: 800001B0
Checksum: 0x46E4
Length: 24
Number of Links: 0

```

例 9-130 显示了 R1 的配置，其中显示了主地址和第二地址在不同的区域内。

例 9-130 R1 的主地址和第二地址配置

```

R1#
interface Ethernet0
ip address 131.108.3.1 255.255.255.0 secondary
ip address 131.108.2.1 255.255.255.0
!

```

(待续)

```
router ospf 1
network 131.108.1.0 0.0.0.255 area 0
network 131.108.2.0 0.0.0.255 area 0
network 131.108.3.0 0.0.0.255 area 2
```

(2) 解决方法

要解决这个问题，可改变 R1 上的配置使得第二地址也属于区域 0。例 9-131 显示了 R1 上的配置，其中显示了第二地址被包含在主地址的区域中。

例 9-131 配置 R1 的第二地址使它与主地址处于同一区域中

```
R1#
interface Ethernet0
ip address 131.108.3.1 255.255.255.0 secondary
ip address 131.108.2.1 255.255.255.0
!
router ospf 1
network 131.108.1.0 0.0.0.255 area 0
network 131.108.2.0 0.0.0.255 area 0
network 131.108.3.0 0.0.0.255 area 0
```

例 9-132 显示了 **show ip route 131.108.3.0** 的输出，表示在解决这个问题之后，路由又回到了路由表中。

例 9-132 确认路由 131.108.3.0 被再次向外通告

```
R2#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
  Known via "ospf 1", distance 110, metric 64, type intra area
  Redistributing via ospf 1
  Last update from 131.108.1.1 on Serial0, 04:22:21 ago
  Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.2.1, 04:22:21 ago, via Serial0
      Route metric is 64, traffic share count is 1
```

9.3.2 问题：OSPF 邻居 (ABR) 不通告汇总路由

当 OSPF 配置了多个区域时，其中一个区域必须是骨干区域。位于骨干区域和任何其它区域边界的路由器都是 ABR。ABR 为一个区域产生汇总 LSA 并将它发送到另一个区域。当 ABR 产生汇总 LSA 失败了，这个区域就与其它区域隔离开了。

这个问题最常见的可能原因如下：

- 区域被配置为一个完全端区。
- ABR 与区域 0 不相连。
- 存在一个不邻接的区域 0。

(待续)

1. OSPF 邻居(ABR)不通告汇总路由——原因: 区域被配置为完全端区

当一个区域被配置为一个端区时, 没有外部 LSA 可以被引入进这个区域。同样地, 一个区域可以被配置为一个完全端区, 这表示没有外部 LSA 或汇总 LSA 可以被引入进这个区域。

图 9-50 显示了用于产生这个问题的 OSPF 网络设置。R1 是一个 ABR, 并且区域 2 被定义为一个完全端区。



图 9-50 用于产生这个问题的网络设置

图 9-51 显示了解决这个问题应遵循的流程图。

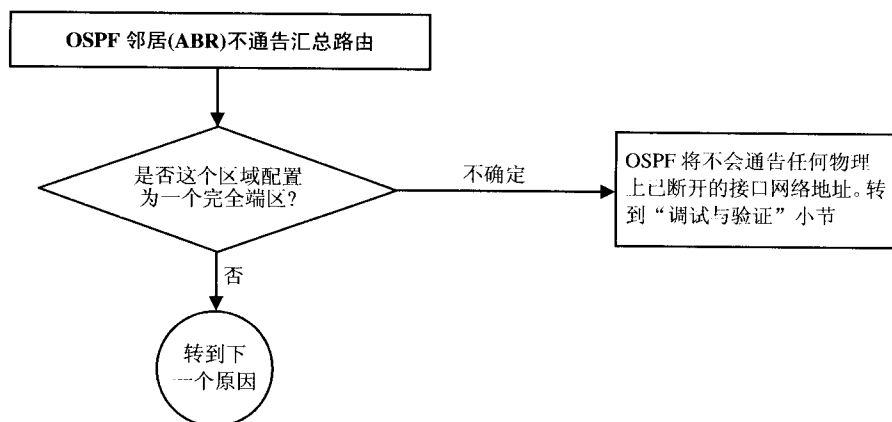


图 9-51 解决问题流程图

(1) 调试与验证

例 9-133 显示了 R1 的配置, 显示区域 2 被配置为一个完全端区域。

例 9-133 R1 的配置显示区域类型

```

R1#
router ospf 1
 network 131.108.1.0 0.0.0.255 area 2
 network 131.108.2.0 0.0.0.255 area 3
 network 131.108.3.0 0.0.0.255 area 0
 area 2 stub no-summary
  
```

例 9-134 显示了对 131.108.2.0 使用 `show ip ospf database summary` 的输出, 表明汇总

LSA 仅在区域 0 产生而没有在区域 2 中产生。

例 9-134 show ip ospf database summary 命令输出显示汇总 LSA 的信息

```
R1#show ospf database summary 131.108.2.0

OSPF Router with ID (131.108.3.1) (Process ID 1)

Summary Net Link States (Area 0)

LS age: 58
Options: (No TOS-capability, DC)
LS Type: Summary Links(Network)
Link State ID: 131.108.2.0 (summary Network Number)
Advertising Router: 131.108.3.1
LS Seq Number: 8000000E
Checksum: 0x4042
Length: 28
Network Mask: /24
TOS: 0 Metric: 1
```

(2) 解决方法

在这个例子中，区域 2 被配置成一个完全端区域，因此没有汇总 LSA 由 ABR 产生传向这个区域。

如果只有一个出口点，就没有必要在一个区域内接收特定的汇总 LSA；缺省的汇总 LSA 的 0.0.0.0 就足够了。在一个完全 NSSA 区域的情形中，也是这样的。

因为这是常见行为，所以对它没有解决方法。可是，如果为这个区域存在多个 ABR 并且需要接收一个特别的汇总 LSA 来避免次优化路由，仅仅从 ABR 上区域 2 的配置中去掉 no-summary 关键字就行了。这将使区域 2 成为一个端区并且所有的汇总 LSA 将被引入进这个区域中。

2. OSPF 邻居(ABR)不通告汇总路由——原因：ABR 没有连接到区域 0

当一个路由器与多个区域相连时，其中一个区域必须是区域 0。如果没有连接到区域 0，ABR 将不会产生汇总 LSA。这是一个标准的 OSPF 行为。

图 9-52 显示了一个遭受这个问题的 OSPF 网络。R1 在区域 2 和区域 3 之间，但它没有连接到区域 0。



图 9-52 OSPF 网络，其中 ABR 没有与区域 0 相连

图 9-53 显示了解决这个问题应遵循的流程图。

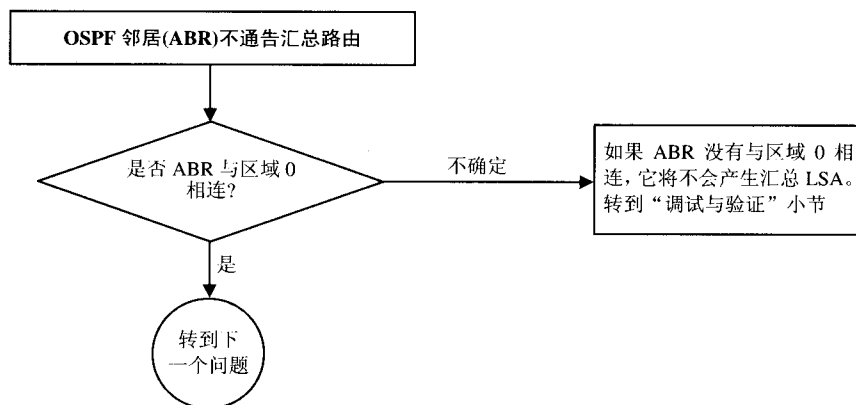


图 9-53 解决问题流程图

(1) 调试与验证

例 9-135 显示 R1 没有产生一个汇总路由。

例 9-135 R1 没有产生一个汇总路由

```

R1#show ip ospf database summary

OSPF Router with ID (131.108.3.1) (Process ID 1)

R1#
  
```

例 9-136 显示了 R1 上的配置, 表明 R1 没有与区域 0 相连。

例 9-136 R1 的配置表明它没有与区域 0 相连

```

router R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 2
network 131.108.3.0 0.0.0.255 area 3
  
```

有另外一种办法可以观察一个区域是否与骨干区域相连。例 9-137 显示了 `show ip ospf` 的输出。如果这个路由器连接到了区域 0, 输出将会显示“它是一个区域边界路由器”。

例 9-137 `show ip ospf` 命令输出表明这个路由器是否为一个 ABR

```

R1#show ip ospf
Routing Process "ospf 1" with ID 131.108.3.1
Supports only single TOS(TOS0) routes
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
  
```

(2) 解决方法

要解决这个问题, 需创建一个骨干区域。在 R1 上创建一个骨干区域之后, 它将开始产生汇总 LSA。

如果在这个网络中不存在区域 0, 必须将其中一个区域转变成区域 0 以使 OSPF 正常工作。

例 9-138 显示了新的配置, 表明从区域 3 中去掉路由 131.108.3.0 并且将它放入区域 0 之后, 现在 R1 已经与区域 0 连接了。

例 9-138 配置 R1 来连接区域 0

```
R1#  
router ospf 1  
network 131.108.1.0 0.0.0.255 area 2  
network 131.108.3.0 0.0.0.255 area 0
```

如果一个骨干区域已经存在, 在最近的 ABR 和 R1 之间创建一条虚链路, 如例 9-139 中所示。

例 9-139 在 R1 和最近的 ABR 之间创建一条虚链路

```
R1#  
router ospf 1  
network 131.108.1.0 0.0.0.255 area 2  
network 131.108.3.0 0.0.0.255 area 3  
area 2 virtual-link 141.108.1.1
```

例 9-140 显示了 **show ip ospf** 的输出, 现在显示 R1 是一个 ABR 了。

例 9-140 确认 R1 是一个 ABR

```
R1#show ip ospf  
Routing Process "ospf 1" with ID 131.108.3.1  
Supports only single TOS(TOS0) routes  
It is an area border router  
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs  
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
```

例 9-141 显示 R1 开始产生汇总 LSA 了。

例 9-141 确认 R1 产生汇总 LSA

```
R1#show ip ospf database summary  
  
OSPF Router with ID (131.108.3.1) (Process ID 1)  
  
Summary Net Link States (Area 0)  
  
LS age: 58  
Options: (No TOS-capability, DC)  
LS Type: Summary Links(Network)  
Link State ID: 131.108.1.0 (summary Network Number)  
Advertising Router: 131.108.3.1  
LS Seq Number: 8000000E  
Checksum: 0x4042  
Length: 28  
Network Mask: /24  
TOS: 0 Metric: 1
```

(待续)

Summary Net Link States (Area 2)

```

LS age: 58
Options: (No TOS-capability, DC)
LS Type: Summary Links(Network)
Link State ID: 131.108.3.0 (summary Network Number)
Advertising Router: 131.108.3.1
LS Seq Number: 8000000E
Checksum: 0x4042
Length: 28
Network Mask: /24
TOS: 0 Metric: 1

```

3. OSPF 邻居 (ABR) 不通告汇总路由——原因: 存在不邻接的区域 0

图 9-54 显示了一个遭受这个问题的 OSPF 网络。R1 和 R2 之间的链路断开了, 这样使得区域 0 不邻接。

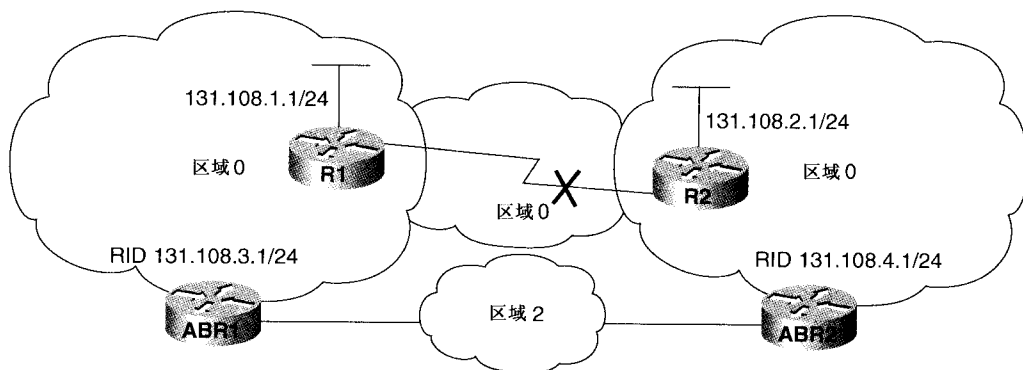


图 9-54 有一个不邻接的区域 0 的 OSPF 网络

图 9-55 显示了解决这个问题应遵循的流程图。

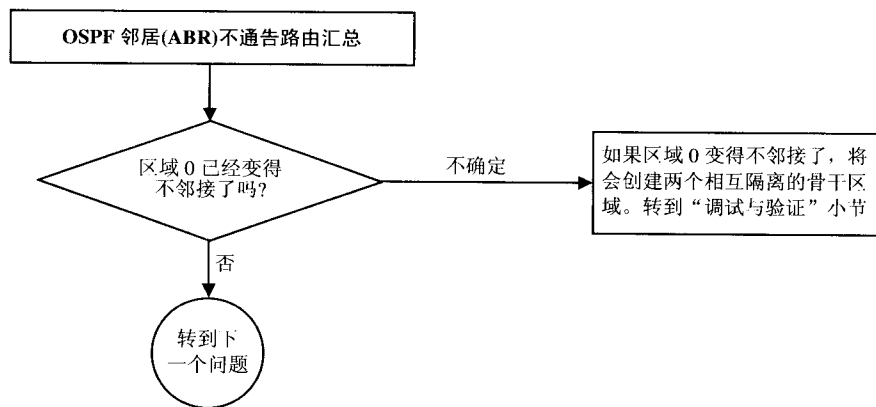


图 9-55 解决问题流程图

(1) 调试与验证

例 9-142 显示了 ABR 2 的路由表。路由 131.108.1.0/24 作为一条区域间路由出现在了路由表中。路由 131.108.1.0 确实存在于骨干区域中, 但是因为骨干区域已经变得不邻接, 所以

这条路由通过区域 2 作为一条区域间路由而传向 ABR 2。ABR 2 将不会为这条路由产生汇总 LSA，因为它是一条区域间路由。

例 9-142 由于存在一个不邻接的骨干区域，所以 ABR 将接收路由 131.108.1.0/24 作为一条区域间路由

```
ABR2#show ip route 131.108.1.0
Routing entry for 131.108.1.0/24
  Known via "ospf 1", distance 110, metric 129, type inter area
  Redistributing via ospf 1
  Last update from 131.108.0.1 on Serial0.1, 00:56:02 ago
  Routing Descriptor Blocks:
    * 131.108.0.1, from 131.108.3.1, 00:56:02 ago, via Serial0.1
      Route metric is 129, traffic share count is 1
```

例 9-143 显示了 ABR 2 没有为这条路由产生一条汇总 LSA 进入区域 0。

例 9-143 没有为 131.108.1.0 产生汇总 LSA 进入区域 0

```
ABR2#show ip ospf database summary 131.108.1.0

OSPF Router with ID (131.108.4.1) (Process ID 1)

ABR2#
```

(2) 解决方法

这显然是一个错误的设计。骨干区域通过一条链路互相连接，如果那条链路出现故障了，骨干区域就会变得不邻接。ABR 2 从 ABR 1 接收区域间路由但是不为这些路由创建汇总 LSA。这与 OSPF RFC 2328 一致，即一条汇总 LSA 不应该为区域间路由而被引入到骨干区域中。

为区域间路由引入一条汇总 LSA 到骨干区域中会使骨干区域相互隔离。要解决这个问题，请确信避免单点失效，如同这个例子中所示，那将使一个骨干区域变得不邻接。另一个解决方法是在 ABR 1 和 ABR 2 之间创建一条虚链路。在创建一条虚链路之后，ABR 1 和 ABR 2 的区域 0 数据库将通过虚链路被同步。就好像 ABR 1 和 ABR 2 之间有了一条物理链路并且这条链路在区域 0 中一样。仅有的不同是 ABR 1 和 ABR 2 都将使用区域 2 来转发 OSPF 分组。要获取关于虚链路的更多信息，请参阅第 8 章。

例 9-144 显示了 ABR 1 和 ABR 2 的配置，显示配置了一条虚链路使得区域 0 变得邻接了。

例 9-144 在 ABR1 和 ABR2 之间配置一条虚链路

```
ABR1#
router ospf 1
 network 131.108.0.0 0.0.0.255 area 2
 network 131.108.3.0 0.0.0.255 area 0
 area 2 virtual-link 131.108.4.1

ABR2#
router ospf 1
 network 131.108.0.0 0.0.0.255 area 2
 network 131.108.4.0 0.0.0.255 area 0
 area 2 virtual-link 131.108.3.1
```

首先通过定义中继区域来配置一条虚链路。这是两个 ABR 之间共同的区域。这个区域被用于在两个 ABR 之间形成一条虚邻接体。在这个例子中是区域 2。在 ABR 1 上，在路由器 OSPF 下定义的命令是 **area 2 virtual-link 131.108.4.1**。地址 131.108.4.1 是 ABR 2 的路由器 ID。类似地，地址 131.108.3.1 是 ABR 1 的路由器 ID。路由器 ID 是路由器上最大的 IP 地址——或者，如果存在一个回环接口，这个回环接口地址将变成路由器 ID。强烈推荐你定义一个回环接口地址这样它将被选举成为一个路由器 ID。一个很好的理由是如果一条链路而不是一个回环接口被选为一个路由器 ID 而且那条链路断开了，那么路由器 ID 将被改变。当一个路由器 ID 改变时，也会断开这条虚链路。另一方面，如果定义了一个回环地址作为一个路由器 ID，路由器 ID 将总是相同的。

例 9-145 显示在创建虚链路之后，ABR 2 开始接收发往包括链路 131.108.1.0/24 的区域 0 的路由器 LSA。

例 9-145 确认 ABR 2 收到了发往区域 0 的路由器 LSA，区域 0 包括链路 131.108.1.0 / 24

```
ABR2#show ip ospf database router 131.108.1.1

OSPF Router with ID (131.108.3.1) (Process ID 1)

Router Link States (Area 0)
Routing Bit Set on this LSA
LS age: 6 (DoNotAge)
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.1.1
Advertising Router: 131.108.1.1
LS Seq Number: 80000002
Checksum: 0xC375
Length: 48
Number of Links: 3

Link connected to: a point-to-point Link
(Link ID) Neighboring Router ID: 131.108.3.1
(Link Data) Router Interface address: 131.108.3.2
Number of TOS metrics: 0
TOS 0 Metrics: 64

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.3.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 1

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.1.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
```

例 9-146 显示在创建虚链路后, R2 开始接收发往 131.108.1.0/24 的区域内路由。

例 9-146 确认 R2 接收发往 131.108.1.0/24 的区域内路由

```
ABR2#show ip route 131.108.1.0
Routing entry for 131.108.1.0/24
  Known via "ospf 1", distance 110, metric 193, type intra area
  Redistributing via ospf 1
  Last update from 131.108.4.1 on Serial0.1, 00:56:02 ago
  Routing Descriptor Blocks:
    * 131.108.4.1, from 131.108.4.1, 00:56:02 ago, via Serial0.1
    Route metric is 193, traffic share count is 1
```

9.3.3 问题: OSPF 邻居不通告外部路由

每当 OSPF 中发生一个路由重发布的时候, 就会产生一个外部 LSA (类型 5), 这个 LSA 被通过 OSPF 网络泛洪。外部 LSA 不会被引入到端区、完全端区和 NSSA 区域中。

这个问题最常见的可能原因如下:

- 这个区域被配置成了端区或者 NSSA 区域。
- NSSA ABR 没有把类型 7 LSA 转换成类型 5 LSA。

1. OSPF 邻居不通告外部路由——原因: 区域被配置成了端区或者 NSSA 区域

在 OSPF 中, 类型 5 LSA 在端区或者 NSSA 区域中是不允许的。当在一个完全处于端区或者 NSSA 区域中的路由器上键入 **redistribute** 命令时, 会显示一个警告消息。这个配置中的 **redistribute** 命令不能够引入任何外部 LSA 进入到一个端区或者 NSSA 区域中。

图 9-56 显示了解决这个问题应遵循的流程图。

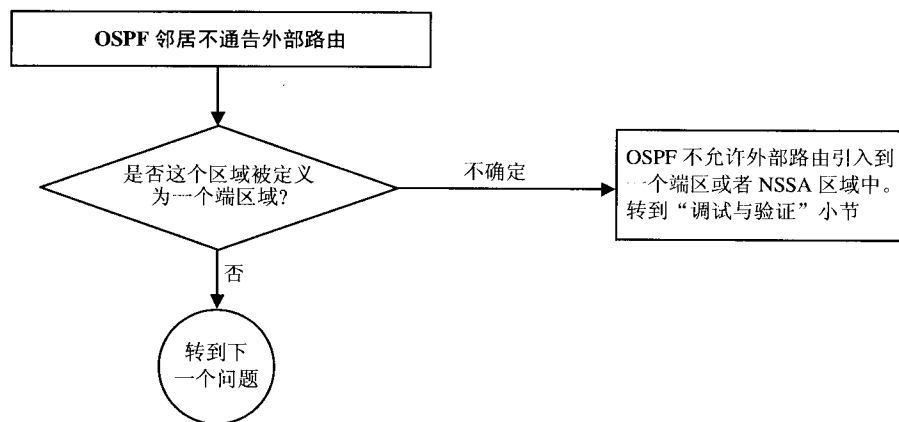


图 9-56 解决问题流程图

(1) 调试与验证

例 9-147 显示了在一个端区中的路由器上尝试从另一种路由协议重发布路由进 OSPF 协议时的配置错误。

例 9-147 在一个端区域路由器上重发布路由进 OSPF 时引起的错误

```
R1(config)#router ospf 1
R1(config-router)#redistribute rip subnets
Warning: Router is currently an ASBR while having only one area which is a stub area
```

例 9-148 显示了 R1 上的配置。尽管 RIP 协议被重发布, R1 也不会为 RIP 子网产生类型 5 LSA, 因为 R1 完全在一个端区域中。要获得更多关于类型 5 LSA 的信息请参阅第 8 章。

例 9-148 当一个 OSPF 区域被定义为一个端区域的时候重发布 RIP 到 OSPF 中

```
R1#
router ospf 1
redistribute rip subnets
network 131.108.1.0 0.0.0.255 area 2
network 131.108.2.0 0.0.0.255 area 2
area 2 stub
```

例 9-149 显示了 R1 没有产生外部 LSA, 因为 R1 完全处于一个端区域中。R1 将忽视 redistribution 命令。

例 9-149 没有由 R1 产生的外部 LSA

```
R1#show ip ospf database external 132.108.3.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

R1#
```

(2) 解决方法

这个问题可以有两种解决办法。一个办法是在区域 2 中所有路由器上去掉 **area 2 stub** 命令从而使得区域 2 成为一个普通区域。这不是一个好的方法, 因为有时, 一个端区域不需要外部 LSA。

第二个解决方法是一个首选的解决方案, 即改变整个区域为一个 NSSA。一个 NSSA 允许外部路由重发布。可是, NSSA 将不会产生类型 5 LSA。它将产生类型 7 LSA, 类型 7 LSA 将被 NSSA ABR 转换成类型 5 LSA。第 8 章详细解释了 NSSA 细节。

例 9-150 显示了转换一个端区域成为一个 NSSA 的配置。在 NSSA 的所有路由器上都必须运行 **area id nssa** 命令。

例 9-150 转换一个端区域为一个 NSSA 来允许外部路由重发布

```
R1#
router ospf 1
redistribute rip subnets
network 131.108.1.0 0.0.0.255 area 2
network 131.108.2.0 0.0.0.255 area 2
area 2 nssa
```

例 9-151 显示了 R1 为两个 RIP 路由产生类型 7 LSA 进入 NSSA 区域。NSSA ABR 进一步转换这些类型 7 LSA 成为类型 5 LSA, 它们能够被泛洪到剩余的 OSPF 区域中。

例 9-151 显示 OSPF 为 RIP 路由产生类型 7 LSA

```
R1#show ip ospf database nssa-external 132.108.3.0
```

```
OSPF Router with ID (131.108.2.1) (Process ID 1)
```

Type-7 AS External Link States (Area 2)

```
LS age: 1161
```

```
Options: (No TOS-capability, Type 7/5 translation, DC)
```

```
LS Type: AS External Link
```

```
Link State ID: 132.108.3.0 (External Network Number )
```

```
Advertising Router: 131.108.2.1
```

```
LS Seq Number: 80000001
```

```
Checksum: 0x550
```

```
Length: 36
```

```
Network Mask: /24
```

```
Metric Type: 2 (Larger than any link state path)
```

```
TOS: 0
```

```
Metric: 1
```

```
Forward Address: 0.0.0.0
```

```
External Route Tag: 1
```

```
R1#
```

```
R1#show ip ospf database nssa-external 132.108.4.0
```

```
OSPF Router with ID (131.108.2.1) (Process ID 1)
```

Type-7 AS External Link States (Area 2)

```
LS age: 1161
```

```
Options: (No TOS-capability, Type 7/5 translation, DC)
```

```
LS Type: AS External Link
```

```
Link State ID: 132.108.4.0 (External Network Number )
```

```
Advertising Router: 131.108.2.1
```

```
LS Seq Number: 80000001
```

```
Checksum: 0x550
```

```
Length: 36
```

```
Network Mask: /24
```

```
Metric Type: 2 (Larger than any link state path)
```

```
TOS: 0
```

```
Metric: 1
```

```
Forward Address: 0.0.0.0
```

```
External Route Tag: 1
```

```
R1#
```

2. OSPF 邻居不通告外部路由——原因: NSSA ABR 不把类型 7 LSA 转换成类型 5 LSA

NSSA RFC 1587 表明在 NSSA ABR 转换类型 7 LSA 成为类型 5 LSA 之前, 所有 NSSA ABR 都必须为一个特殊的 NSSA 检查所有的 NSSA ABR。有着最大路由器 ID 的 ABR 必须进行把类型 7 转换成类型 5 的工作。如果类型 7 没有被 NSSA ABR 转换成类型 5, 那么将不会有 NSSA 旁边的路由器注意到在 NSSA 区域中发生的外部路由重发布过程。这使得 NSSA 的整个计划落空。

图 9-57 显示了一个遭受这个问题的 OSPF 网络。路由器 2 是一个 NSSA ABR, 但它没有转换类型 7 LSA 成为类型 5 LSA。

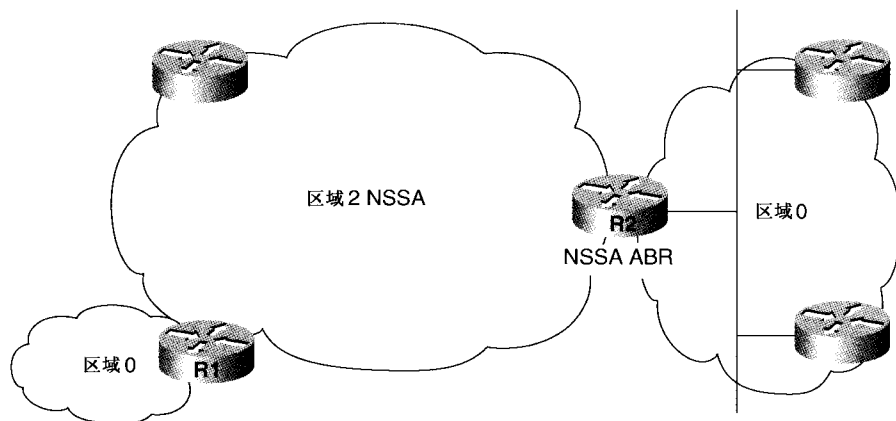


图 9-57 OSPF 网络, 其中 OSPF 邻居 R2 没有通告外部路由到区域 0 中

图 9-58 显示了解决这个问题应遵循的流程图。

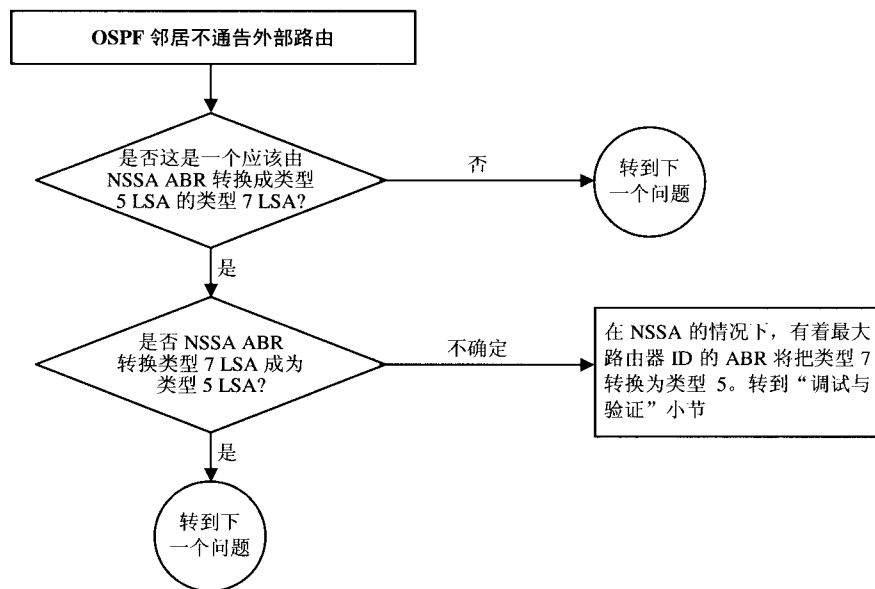


图 9-58 解决问题流程图

成为类型 5

所有 NSSA

ABR 必须

, 那么将不

使得 NSSA

, 但它没有

(1) 调试与验证

例 9-152 显示了在路由 132.108.4.0 上 **show ip ospf database nssa-external** 的输出, 表示 R2 产生类型 7 LSA。

例 9-152 **show ip ospf database nssa-external** 命令输出表示类型 7 LSA 被产生出来了

```
R2#show ip ospf database nssa-external 132.108.4.0

OSPF Router with ID (131.108.1.2) (Process ID 1)

Type-7 AS External Link States (Area 2)
LS age: 1161
Options: (No TOS-capability, Type 7/5 translation, DC)
LS Type: AS External Link
Link State ID: 132.108.4.0 (External Network Number)
Advertising Router: 131.108.1.2
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 1
```

例 9-153 显示了为 132.108.4.0 使用的 **show ip ospf database external** 命令的输出, 表示 R2 没有转换类型 7 LSA 成为类型 5 LSA。

例 9-153 **show ip ospf database external** 命令输出表示类型 7 LSA 到类型 5 LSA 的转换没有发生

```
R2#show ip ospf database external 132.108.4.0

OSPF Router with ID (131.108.1.2) (Process ID 1)

R2#
```

例 9-154 显示了 R2 是一个 NSSA ABR, 因此假定由它来执行转换。

例 9-154 确认 R2 是一个 NSSA ABR, 它要将类型 7 转换到类型 5

```
R2#show ip ospf
Routing Process "ospf 1" with ID 131.108.1.2
Supports only single TOS(TOS0) routes
It is an area border router
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 3. Checksum Sum 0x14DAA
```

(待续)

```

Number of DCbitless external LSA 0
Number of DoNotAge external LSA 0
Number of areas in this router is 4. 3 normal 0 stub 1 nssa
  Area BACKBONE(0)
    Number of interfaces in this area is 2
    Area has no authentication
    SPF algorithm executed 60 times
    Area ranges are
    Number of LSA 16. Checksum Sum 0x9360D
    Number of DCbitless LSA 7
    Number of indication LSA 0
    Number of DoNotAge LSA 0
  Area 2
    Number of interfaces in this area is 1
    It is a NSSA area
    Perform type-7/type-5 LSA translation
    Area has no authentication
    SPF algorithm executed 54 times
    Area ranges are
    Number of LSA 11. Checksum Sum 0x7A449
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0

```

例 9-155 显示在 NSSA 区域中的另一个路由器，即 R1，它也声称自己是一个 ABR。这个路由器是一个假的 ABR 因为它不与区域 0 相连接。

例 9-155 show ip ospf 命令输出表明 R1 声称自己是一个 ABR

```

R1#show ip ospf
Routing Process "ospf 1" with ID 131.108.2.1
Supports only single TOS(TOS0) routes
It is an area border router
Summary Link update interval is 00:30:00 and the update due in 00:29:48
External Link update interval is 00:30:00 and the update due in 00:19:43
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Number of DCbitless external LSA 0
Number of DoNotAge external LSA 0
Number of areas in this router is 2. 1 normal 0 stub 1 nssa
  Area BACKBONE(0) (Inactive)
    Number of interfaces in this area is 1
    Area has no authentication
    SPF algorithm executed 2 times
    Area ranges are
    Link State Update Interval is 00:30:00 and due in 00:29:47
    Link State Age Interval is 00:20:00 and due in 00:19:47
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
  Area 2

```

(待续)

```
Number of interfaces in this area is 1
It is a NSSA area
Area has no authentication
SPF algorithm executed 65 times
Area ranges are
Link State Update Interval is 00:30:00 and due in 00:16:27
Link State Age Interval is 00:20:00 and due in 00:14:39
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
```

例 9-156 显示了 R1 代替 R2 完成了全部转换工作。

例 9-156 R1 代替 R2 将类型 7 LSA 转换到类型 5 LSA

```
R1#show ip ospf database external 132.108.3.0
```

```
OSPF Router with ID (131.108.2.1) (Process ID 1)
```

Type-5 AS External Link States

```
LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.3.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 1
```

```
R1#
```

```
R1#show ip ospf database external 132.108.4.0
```

```
OSPF Router with ID (131.108.2.1) (Process ID 1)
```

Type-5 AS External Link States

```
LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.4.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
```

(待续)

ABR。这

3

待续)

```
Checksum: 0x550
Length: 36
Network Mask: /24
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 1
R1#
```

例 9-157 显示了 R1 上的 OSPF 配置。在 R1 上的一个 **network** 语句包含 R1 的一个在区域 0 中的接口。**network** 语句产生了问题，因为 R1 上的接口都不属于骨干区域。

例 9-157 R1 的配置显示一个接口在区域 0 中

```
R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 2
network 131.108.2.0 0.0.0.255 area 2
network 131.108.5.0 0.0.0.255 area 0
area 2 nssa
```

(2) 解决方法

在这种情况下，R1 有一个路由器 ID(RID)131.108.2.1，而 R2 有一个路由器 ID131.108.1.2，如例 9-154 和例 9-155 的输出所示。因为 R1 有一个比 R2 大的 RID，所以 R1 进行转换工作因而类型 5 LSA 进入了一个黑洞。

你可以使用多种方法来解决这个问题：

- 去掉错误的 **network** 语句。
 - 改变 R1 的路由器 ID 使它小于 R2 的路由器 ID。
 - 改变 R2 的路由器 ID 使它大于 R1 的路由器 ID。
- 例 9-158 显示了最简单的解决方法，即去掉不正确的 **network** 语句。

例 9-158 在 R1 上纠正 **network** 语句以使 R2 执行类型 7 LSA 到类型 5 LSA 的转换

```
R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 2
network 131.108.2.0 0.0.0.255 area 2
no network 131.108.5.0 0.0.0.255 area 0
area 2 nssa
```

其它的解决方法是改变 R1 或者 R2 的路由器 ID，使得 R1 的路由器 ID 比 R2 的小或者 R2 的路由器 ID 比 R1 的大。要改变路由器 ID，可给 R1 的回环接口配置一个比 R2 回环接口小的 IP 地址，接着重启 OSPF 进程。重启 OSPF 将导致网络中断很少的几秒钟时间。你键入 OSPF 配置到路由器中的速度越快，花费的中断时间也就越少。在 Cisco IOS 软件版本 12.0 及更高版本中，已经增加了一个命令可用于手动配置路由器 ID 然后再重启 OSPF。例 9-159 显示了如何在 Cisco IOS 软件版本 12.0 中改变路由器 ID。

例 9-159 改变 R1 的路由器 ID 使它比 R2 的小

```
R2(config)#router ospf 1
R2(config-router)#router-id 131.108.0.1
R2(config-router)#end
R2#clear ip ospf process
```

例 9-160 显示了在修改这个配置后, R2 开始正常地转换类型 7 LSA 成为类型 5 LSA。在 R2 上 show ip ospf database external 的输出显示产生了类型 5 LSA 并且 R1 正在接收类型 5 LSA。

例 9-160 确认 LSA 类型转换问题已经解决了

```
R2#show ip ospf database external 132.108.3.0
      OSPF Router with ID (131.108.1.2) (Process ID 1)

+
Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.3.0 (External Network Number )
Advertising Router: 131.108.1.2
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 1
```

R1#

```
R1#show ip ospf database external 132.108.4.0
      OSPF Router with ID (131.108.1.2) (Process ID 1)
```

```

Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.4.0 (External Network Number )
Advertising Router: 131.108.1.2
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
```

(待续)

```
Forward Address: 0.0.0.0
External Route Tag: 1
```

R1#

9.3.4 问题：OSPF 邻居不通告缺省路由

有时，OSPF 对 OSPF 不知道的目的地地址使用缺省路由。大多数这些时候，这些目的地是 OSPF 外部的网络。不需要引入所有的外部路由到 OSPF 中，而仅仅需要一条用于所有未知外部目的地地址的缺省路由。当缺少这样一条缺省路由的时候，所有的通往任何未知地址的流量都将被 OSPF 丢弃。

OSPF 路由器不通告缺省路由的最常见可能原因如下：

- 缺少 **default-information originate** 命令。
- 在邻居的路由表中缺少缺省路由。
- 邻居试图产生缺省路由进入端区域。
- NSSA 区域中的 ABR/ASBR 不产生类型 7 缺省路由。

1. OSPF 邻居不通告缺省路由——原因：缺少 default-information originate 命令

除非命令 **default-information originate** 出现在 OSPF 配置中，否则 OSPF 不会产生缺省路由。这个命令在配置了它的路由器中产生缺省路由。在 OSPF 中没有其它办法产生缺省路由。图 9-59 显示了一个导致这个问题的网络设置。

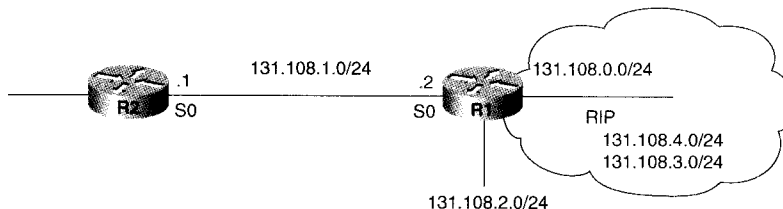


图 9-59 导致这个问题的网络设置

图 9-60 显示了解决这个问题应遵循的流程图。

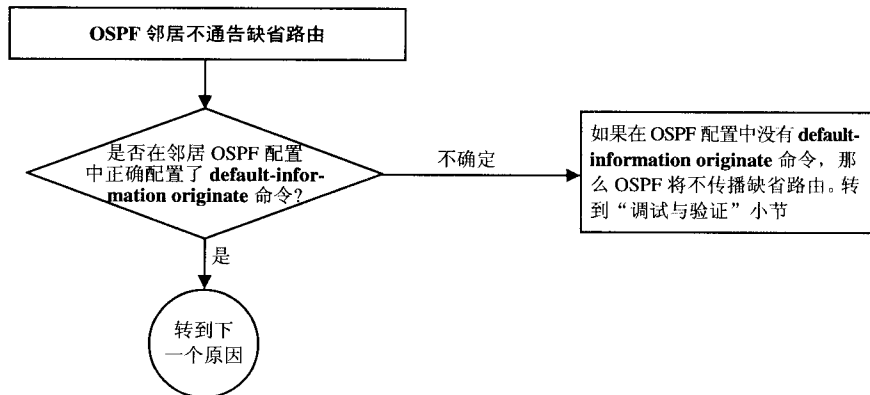


图 9-60 解决问题流程图

(1) 调试与验证

例 9-161 显示了 R1 通过 RIP 接收了一条缺省路由。这是很重要的，因为如果 R1（它是缺省路由的发出者）没有这条缺省路由，它就不能在 OSPF 中产生一条缺省路由。

例 9-161 R1 通过 RIP 接收了一条缺省路由

```
R1#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "rip", distance 120, metric 1, candidate default path
  Redistributing via rip
  Last update from 132.108.0.2 on Serial0, 00:00:16 ago
  Routing Descriptor Blocks:
    132.108.0.2, from 132.108.0.2, 00:00:16 ago, via Serial0
      Route metric is 1, traffic share count is 1
```

例 9-162 显示了 R1 的配置，表明 R1 重发布所有的 RIP 路由到 OSPF 中，就如在这个问题开头提到的，在 OSPF 中产生缺省路由的唯一方式就是通过 **default-information originate** 命令。因此，重发布一条缺省路由将不会引起 OSPF 产生缺省路由。

例 9-162 R1 重发布所有的 RIP 路由到 OSPF 中

```
R1#
router ospf 1
  redistribute rip subnets
  network 131.108.1.0 0.0.0.255 area 0
  network 131.108.2.0 0.0.0.255 area 0
  !
router rip
  network 132.108.0.0
  !
```

例 9-163 显示了 0.0.0.0 不在 R1 的数据库中，但是其它的 RIP 路由在数据库中。这表明是 RIP 的一部分的缺省路由没有重发布进 OSPF 中。

例 9-163 R1 的 OSPF 数据库缺少关于路由 0.0.0.0 的信息

```
R1#show ip ospf database external 0.0.0.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

R1#

R1#show ip ospf database external 132.108.3.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
```

(待续)

```

LS Type: AS External Link
Link State ID: 132.108.3.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 1

```

R1#

R1#show ip ospf database external 132.108.4.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

Type-5 AS External Link States

```

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.4.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 1

```

例 9-164 显示了 R1 的配置，表明在配置中缺少 **default-information originate** 命令。

例 9-164 R1 的配置缺少 **default-information originate** 命令

```

R1#
router ospf 1
 redistribute rip subnets
 network 131.108.1.0 0.0.0.255 area 0
 network 131.108.2.0 0.0.0.255 area 0

```

(2) 解决方法

在这个例子中，即使 R1 从 RIP 接收了一条缺省路由，它也不会重发布到 OSPF 数据库中。

要解决这个问题，需在 R1 上 **router ospf** 选项下配置 **default-information originate** 命令。

例 9-165 显示了改变后配置以解决这个问题。

例 9-165 配置 R1 来包含 default-information originate 命令

```
R1#
router ospf 1
 redistribute rip subnets
 network 131.108.1.0 0.0.0.255 area 0
 network 131.108.2.0 0.0.0.255 area 0
 default-information originate
```

例 9-166 显示在修改了 R1 上的配置之后, R1 正在产生缺省路由到数据库中。

例 9-166 R1 上的新配置解决了缺省路由问题

```
R1#show ip ospf database external 0.0.0.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 0.0.0.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /0
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 1
```

例 9-167 显示在改变了 R1 上的配置之后, R2 开始从 R1 接收缺省路由了。

例 9-167 确认 R2 从 R1 接收缺省路由

```
R2#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 1, candidate default path
  Tag 1, type extern 2, forward metric 128
  Redistributing via ospf 1
  Last update from 131.108.1.2 on Serial0, 00:54:59 ago
  Routing Descriptor Blocks:
    * 131.108.1.2, from 131.108.2.1, 00:54:59 ago, via Serial0
      Route metric is 1, traffic share count is 1
```

2. OSPF 邻居不通告缺省路由——原因: 在邻居的路由表中缺少缺省路由

除非在路由器的 OSPF 中配置了 **default-information originate** 命令, 否则 OSPF 将不产生缺省路由。这个命令有一个更进一步的限制: 路由器必须在它的路由表中有一条缺省路由

命令。

数据库中。
inate 命令。

来产生一条缺省路由。

图 9-61 显示了解决这个问题应遵循的流程图。

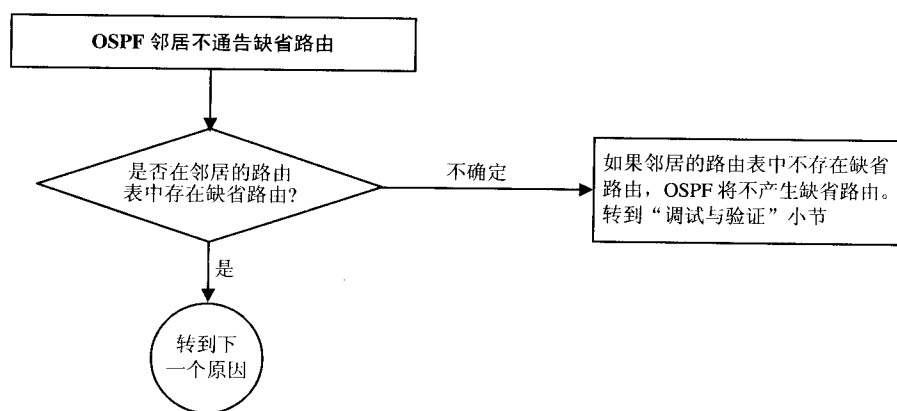


图 9-61 解决问题流程图

(1) 调试与验证

例 9-168 显示了 R1 没有在 OSPF 数据库中产生缺省路由, 因为在路由表中没有缺省路由。参见图 9-56 的网络设置。

例 9-168 R1 不产生缺省路由

```

R1#show ip route 0.0.0.0
% Network not in table
R1#

R1#show ip ospf database external 0.0.0.0
      OSPF Router with ID (131.108.2.1) (Process ID 1)

R1#
  
```

例 9-169 显示了 R1 上的配置以确认命令 **default-information originate** 也已经配置了来消除在前面小节提到的原因。

例 9-169 确认 R1 的配置包括了 default-information originate 命令

```

R1#
router ospf 1
 redistribute rip subnets
 network 131.108.1.0 0.0.0.255 area 0
 network 131.108.2.0 0.0.0.255 area 0
 default-information originate
  
```

(2) 解决方法

这个问题可以通过两种方式解决。一种是确信缺省路由出现在了路由表中。这些信息可以从任何其它路由协议中导出, 包括静态或者缺省路由信息。

例 9-170 显示了在缺省路由回到 R1 的路由表中之后, R1 开始产生缺省外部 LSA。

例 9-170 R1 中的缺省路由回来了，因而它为 0.0.0.0 产生了外部 LSA

```

R1#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "rip", distance 120, metric 1, candidate default path
  Redistributing via rip
  Last update from 132.108.1.1 on Serial1, 00:00:16 ago
  Routing Descriptor Blocks:
    132.108.1.1, from 132.108.1.1, 00:00:16 ago, via Serial0
    Route metric is 1, traffic share count is 1

R1#show ip ospf database external 0.0.0.0

      OSPF Router with ID (131.108.2.1) (Process ID 1)

Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 0.0.0.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /0
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 1

```

第二种方式是伴随 **default-information originate** 命令使用 **always** 关键字。Always 关键字命令路由器不论在路由表中有或者没有缺省路由的情况下都要产生缺省路由。

例 9-171 显示了 R1 的配置，R1 在 OSPF 数据库中产生缺省路由。

例 9-171 配置 always 关键字强迫产生缺省路由

```

R1#
router ospf 1
 redistribute rip subnets
 network 131.108.1.0 0.0.0.255 area 0
 network 131.108.2.0 0.0.0.255 area 0
 default-information originate always

```

例 9-172 显示了在 R1 的 OSPF 数据库中产生了缺省路由。

例 9-172 确认 R1 产生缺省路由

```

R1#show ip ospf database external 0.0.0.0

      OSPF Router with ID (131.108.2.1) (Process ID 1)

```

(待续)

Type-5 AS External Link States

```

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 0.0.0.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /0
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 1

```

例 9-173 显示在改变了 R1 上的配置后, R2 开始接收从 R1 来的缺省路由。

例 9-173 R2 从 R1 接收缺省路由

```

R2#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
    Known via "ospf 1", distance 110, metric 1, candidate default path
    Tag 1, type extern 2, forward metric 128
    Redistributing via ospf 1
    Last update from 131.108.1.2 on Serial0, 00:54:59 ago
    Routing Descriptor Blocks:
    * 131.108.1.2, from 131.108.2.1, 00:54:59 ago, via Serial0
      Route metric is 1, traffic share count is 1

```

3. OSPF 邻居不通告缺省路由——原因: 邻居试图产生缺省路由进入端区域

当一个区域定义为端区域的时候, 那么在它中间不允许外部 LSA。这包括由一个路由器的 **default-information originate** 命令创建的类型 5 缺省外部 LSA。一个 ABR 在一个端区域中自动产生一条类型 3 汇总 LSA 缺省路由; 可是, 如果在 ABR 或者非 ABR 上配置了 **default-information originate** 命令, 它将被作为缺省路由通告, 因为这个命令产生类型 5 LSA, 但是类型 5 LSA 不允许存在于一个端区域中。在图 9-62 中, R1 是一个完全在区域 2 之中的路由器, 区域 2 被定义为一个端区域。

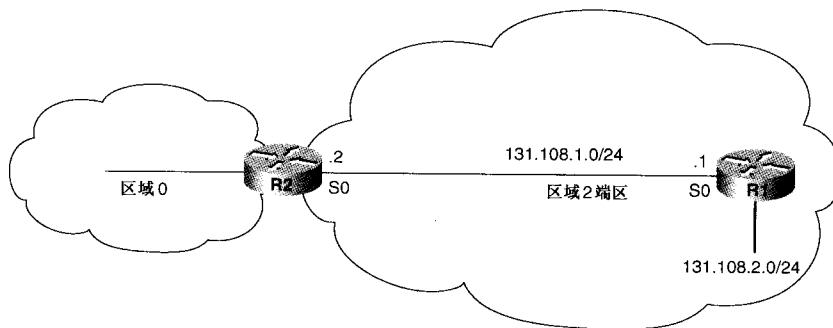


图 9-62 产生这个问题的网络设置

图 9-

(1)

例 9-

候得到的
使用命令

例 9-

警告消息

R1(c
R1(c
Warn

例 9-

可是, R1
数据库中也不

例 9-

R1#
route
net
net
net
def
area
R1#
R1#

(2) 角
在这个

图 9-63 显示了解决这个问题的流程图。

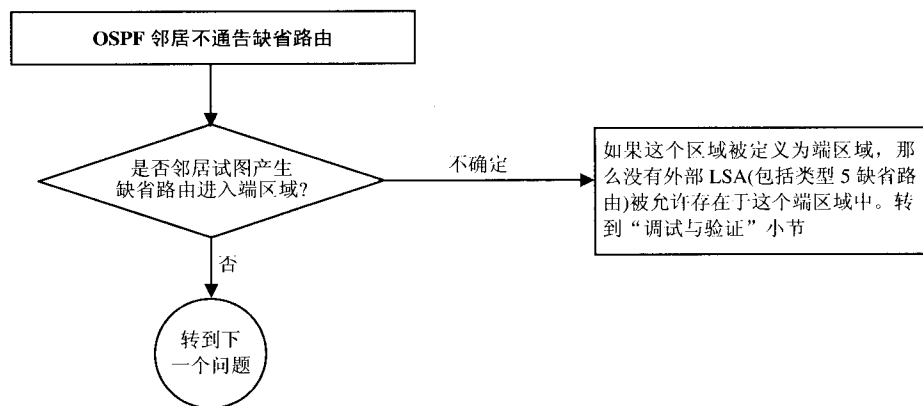


图 9-63 解决问题流程图

(1) 调试与验证

例 9-174 显示了当 R1 试图用 **default-information originate** 命令产生一条缺省路由的时候得到的警告消息。R1 的所有接口都在区域 2 中，区域 2 定义为一个端区域，这样 R1 不能使用命令 **default-information originate**，因为它将使得 R1 成为一个 ASBR。

例 9-174 当试图用 **default-information originate** 命令产生一条缺省路由的时候产生的警告消息

```

R1(config)#router ospf 1
R1(config-router)#default-information originate
Warning: Router is currently an ASBR while having only one area which is a stub area
  
```

例 9-175 显示了在 R1 上的配置，揭示了 R1 配置有 **default-information originate** 命令。可是，R1 将不产生缺省的类型 5 LSA 到区域 2 中，因为它是一个端区域。在 R1 的 OSPF 数据库中也不能找到有关 0.0.0.0 的信息。

例 9-175 R1 的配置表明它定义为一个端区域

```

R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 2
network 131.108.2.0 0.0.0.255 area 2
default-information originate
area 2 stub

R1#show ip ospf database external 0.0.0.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

R1#
  
```

(2) 解决方法

在这个例子中，R1 的所有接口都包括在区域 2 中，区域 2 是一个端区域。当配置了

default-information originate 命令时，它简单地用一条警告消息忽略这个区域。

通常，当一个区域定义成一个端区域的时候，由 ABR 产生一个缺省类型 3 汇总。

这个问题可以有 3 种解决方式：

- 改变区域为 NSSA 类型，然后产生一条缺省的类型 7 LSA。
- 改变端区域使它成为一个普通区域，接着产生一条缺省路由。
- 配置一个静态缺省路由。

第一种解决方式在本节后面讨论。第二种方式不是一个好办法，因为区域 2 将变成一个普通区域，并且会包含不需要的类型 5 LSA。可是在这种情况下，它是一个可选的解决方案。

例 9-176 显示了要达到这个目的所需要的配置改变。

例 9-176 配置 R1 作为一个普通区域并产生一条缺省路由

```
R1#
router ospf 1
 network 131.108.1.0 0.0.0.255 area 2
 network 131.108.2.0 0.0.0.255 area 2
 default-information originate
 no area 2 stub
```

例 9-177 显示了在配置改变之后 R2 从 R1 接收缺省路由。

例 9-177 确认重配置 R1 成为一个普通区域解决了这个问题

```
R2#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 110, metric 1, candidate default path
  Tag 1, type extern 2, forward metric 128
  Redistributing via ospf 1
  Last update from 131.108.1.1 on Serial0, 00:54:59 ago
  Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.2.1, 00:54:59 ago, via Serial0
      Route metric is 1, traffic share count is 1
```

第三个解决方案实现起来很简单。所需要的是不理睬从 ABR 来的汇总缺省路由以防这个区域是一个端区域。如果希望得到那条汇总缺省路由，就不需要在配置中有任何改变。

例 9-178 显示了 R1 配置有静态的缺省路由。为了忽视在一个区域中所有路由器上的汇总缺省路由，必须在这个区域中所有路由器上配置静态缺省路由。这使得这种解决方式不受欢迎。

例 9-178 R1 配置有静态缺省路由

```
R1(config)#ip route 0.0.0.0 0.0.0.0 131.108.2.2
R1#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "static", distance 1, metric 0, candidate default path
  Routing Descriptor Blocks:
    * 131.108.2.2
      Route metric is 0, traffic share count is 1
```

4. OSPF 邻居不通告缺省路由——原因: NSSA 区域中的 ABR/ASBR 不产生类型 7 缺省路由

当定义了一个 NSSA 的时候, 缺省时, NSSA ABR 不产生任何缺省路由。这与在端区域或者完全端区域中的情形不同。当一个区域被定义成端区域, 则 ABR 以汇总 LSA 的形式产生类型 3 缺省路由。这是因为端区域中不能够有任何类型 5 LSA, 因此缺省路由一定不是类型 5 LSA。这在完全端区域中也是一样的。

当一个 NSSA 区域定义有 **no-summary** 选项时, NSSA ABR 自动产生一个类型 3 缺省汇总路由。这创建了一个完全 NSSA。在图 9-64 中, R1 是一个 NSSA ASBR, 它试图产生缺省路由到区域 2 中, 区域 2 是一个 NSSA 区域。

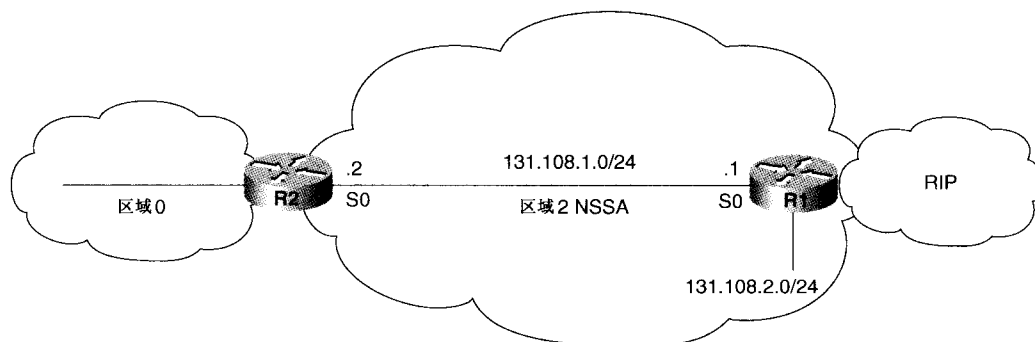


图 9-64 导致这个问题的网络设置

图 9-65 显示了解决这个问题应遵循的流程图。

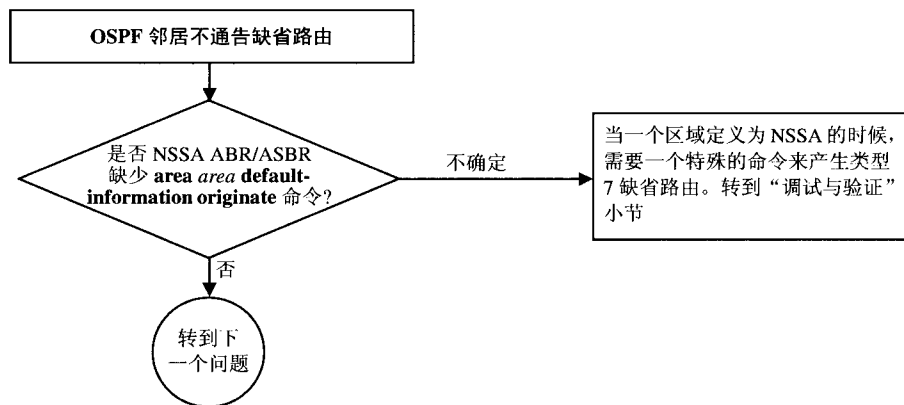


图 9-65 解决问题流程图

(1) 调试与验证

例 9-179 显示了 R1 上的配置。配置显示 R1 试图产生一条缺省路由。

例 9-179 R1 被错误配置产生缺省路由

```
R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 2
```

(待续)

```
network 131.108.2.0 0.0.0.255 area 2
default-information originate
area 2 nssa
```

例 9-180 显示 R1 不产生缺省路由。

例 9-180 R1 不产生缺省路由

```
R1#show ip ospf database external 0.0.0.0
      OSPF Router with ID (131.108.2.1) (Process ID 1)

R1#

R1#show ip ospf database nssa-external 0.0.0.0

      OSPF Router with ID (131.108.2.1) (Process ID 1)

R1#
```

(2) 解决方法

键入在例 9-181 中加亮的命令来在 NSSA 中产生缺省路由。这个命令在 NSSA ABR 或者 NSSA ASBR 中都有效。

例 9-181 在 NSSA 中产生缺省路由

```
R1#
router ospf 1
network 131.108.1.0 0.0.0.255 area 2
network 131.108.2.0 0.0.0.255 area 2
area 2 nssa default-information originate
```

在过去, 这个命令仅仅在 NSSA ABR 上有效, 但是 Cisco IOS 软件版本 12.0.11 和 12.1.2 也提供了支持, 使它在 NSSA ASBR 上可以使用。

例 9-182 显示在配置了这个命令之后, R2 从 R1 收到了缺省路由。

例 9-182 R1 成功产生了缺省路由

```
R2#show ip route 0.0.0.0
Routing entry for 0.0.0.0/0, supernet
  Known via "ospf 1", distance 120, metric 1, candidate default path, type NSSA external
  2,
  forward metric 64
  Redistributing via ospf 1
  Last update from 131.108.1.1 on Serial0, 00:00:03 ago
  Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.2.1, 00:00:03 ago, via Serial0
      Route metric is 1, traffic share count is 1
```

9.4 OSPF 路由安装疑难解析

本节讨论与路由安装有关的问题。这表示 OSPF 路由器与它们的邻居已经完全同步了它们的数据库，但是没有安装路由到路由表中。

在路由进入数据库之后，路由没有安装到路由表可能有多个原因。本节详细讨论这些原因同时也给出了如何解决这些问题的办法。

OSPF 没有成功安装路由到路由表中最常见原因如下：

- OSPF 不在路由表中安装任何路由。
- OSPF 不在路由表中安装外部路由。

9.4.1 问题：OSPF 不在路由表中安装任何路由

发现路由在数据库中但却不在路由表中，这也是一个 OSPF 中常见的问题。当 OSPF 发现数据库中存在任意的不一致的时候，它不会在路由表中安装任何路由。本节中我们假设发送者通告了数据库中的路由。如果发送者没有通告路由，或者如果路由甚至没有在数据库表中出现，就得首先解决这个问题。对于 OSPF 不通告路由的疑难问题已经在前面的章节中讨论过了。

这个问题最常见的可能原因如下：

- 网络类型不匹配。
- IP 地址在通过双重串行链路（dual serial-connected）连接的路由器上配反了（flipped）或者发生了一个子网 / 掩码不匹配错误。
- 一边是有编号的而另一边是无编号的点到点链路。
- 分发表阻止安装路由。
- 在一个广播网络类型的完全网状互连帧中继网络中有中断的 PVC。

图 9-66 显示了导致 OSPF 路由安装问题的网络设置。中间的网络云是没有关系的。它可以是帧中继、PPP HDLC 或者一些其它类型，但在这里，它必须是一条点到点 WAN 链路。



图 9-66 用于产生路由安装问题的 OSPF 网络设置

例 9-183 显示了 R2 没有在路由表中安装任何路由。

例 9-183 R2 在它的路由表中没有路由

```
R2#show ip route ospf
R2#
```

1. OSPF 没有在路由表中安装任何路由——原因：网络类型不匹配

一个不匹配的网络类型在数据库中产生了一致，因而 OSPF 不会在路由表中安装这些

路由。这种情形在 NBMA 网络中很常见，一边是一个点到点网络类型而另一边是一个广播网络类型。这个问题当一边被定义成点到多点网络类型而另一边还是非广播类型的时候也会发生。

在这个例子中，一边被定义成广播类型而另一边被定义成点到点类型。当一个接口网络类型被定义成广播类型的时候，OSPF 认为那条链路是一条中继链路并且将那条链路作为一条中继链路放入到路由器 LSA 中。

图 9-67 显示了解决问题应遵循的流程图。

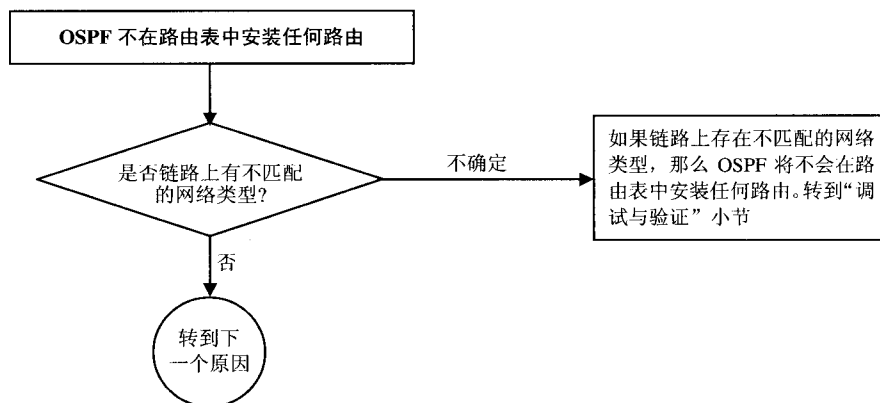


图 9-67 解决问题流程图

(1) 调试与验证

例 9-184 显示了在 R1 和 R2 上的接口配置。R1 串行接口网络类型为广播类型，而 R2 使用缺省网络类型，即非广播类型。

例 9-184 R1 和 R2 的网络类型

```

R1#
interface Serial0
 ip address 131.108.1.1 255.255.255.0
 ip ospf network broadcast
!

R2#
interface Serial0
 ip address 131.108.1.2 255.255.255.0
  
```

例 9-185 显示了在两个路由器上的 Serial 0 接口上 `show ip ospf interface` 的输出，表明在两端有不匹配的网络类型。

例 9-185 确定在 Serial 0 接口上是否 R1 和 R2 有不匹配的网络类型

```

R1#show ip ospf interface serial 0

Serial0 is up, line protocol is up
 Internet Address 131.108.1.1/24, Area 0
 Process ID 20, Router ID 131.108.2.1, Network Type BROADCAST, Cost: 64
  
```

(待续)

```

Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 131.108.2.1, Interface address 131.108.1.1
Backup Designated router (ID) 131.108.2.2, Interface address 131.108.2.2
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 00:00:08
Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 131.108.2.2 (Backup Designated Router)
Suppress hello for 0 neighbor(s)

```

```

R2#show ip ospf interface serial 0
Serial0 is up, line protocol is up
Internet Address 131.108.1.2/24, Area 0
  Process ID 20, Router ID 131.108.1.2, Network Type POINT_TO_POINT, Cost: 64
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 00:00:02
Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 131.108.2.1
Suppress hello for 0 neighbor(s)

```

例 9-186 显示了在每个路由器上路由器 LSA 的输出。路由器 LSA 报错说通告路由器不可达。这就是为什么路由器没有在路由表中安装路由的原因。

例 9-186 发往 R1 和 R2 的 LSA 表示通告路由器不可达

```

R1#show ip ospf database router 131.108.1.2

Adv Router is not-reachable
LS age: 418
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.1.2
Advertising Router: 131.108.1.2
LS Seq Number: 80000002
Checksum: 0xFA63
Length: 60
Number of Links: 3
  Link connected to: another Router (point-to-point)
    (Link ID) Neighboring Router ID: 131.108.2.1
    (Link Data) Router Interface address: 131.108.1.2
  Number of TOS metrics: 0
  TOS 0 Metrics: 64

  Link connected to: a Stub Network
    (Link ID) Network/subnet number: 131.108.1.0
    (Link Data) Network Mask: 255.255.255.0
  Number of TOS metrics: 0
  TOS 0 Metrics: 64

  Link connected to: a Stub Network
    (Link ID) Network/subnet number: 131.108.0.0

```

(待续)

是一个广播网
的时候也会发

一个接口网络
链路作为一

各
路
调

型, 而 R2 使

输出, 表明在

64

(待续)

```
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

```
R2#show ip ospf database router 131.108.2.1
```

```
Adv Router is not-reachable
LS age: 357
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.2.1
Advertising Router: 131.108.2.1
LS Seq Number: 8000000A
Checksum: 0xD4AA
Length: 48
Number of Links: 2

Link connected to: a Transit Network
(Link ID) Designated Router address: 131.108.1.1
(Link Data) Router Interface address: 131.108.1.1
Number of TOS metrics: 0
TOS 0 Metrics: 64

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.2.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

(2) 解决方法

在这个例子中，在数据库中，链路的一边显示为点到点链路类型，而同一链路的另一边显示为中继链路类型。这导致了数据库中存在不一致，因而路由器将不会在路由表中安装任何路由。

要解决这个问题，可把 R1 的网络类型改回到缺省状态，即点到点类型。例 9-187 显示了如何在 R1 上将网络类型改回到点到点类型。

例 9-187 改变网络类型回到点到点状态

```
R1#
interface Serial0
ip address 131.108.1.1 255.255.255.0
no ip ospf network broadcast
```

例 9-188 显示了串行接口上运行 **show ip ospf interface** 的输出。它表明网络类型是点到点类型。

例 9-188 确认 R1 的 serial 0 接口是点到点网络类型

```
R1#show ip ospf interface serial 0
Serial0 is up, line protocol is up
```

(待续)

```

Internet Address 131.108.1.1/24, Area 0
Process ID 20, Router ID 131.108.2.1, Network Type POINT_TO_POINT, Cost: 64
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:02
Neighbor Count is 1, Adjacent neighbor count is 1
Adjacent with neighbor 131.108.1.2
Suppress hello for 0 neighbor(s)

```

例 9-189 显示了 R2 开始在它的路由表中安装 OSPF 路由。

例 9-189 确认 OSPF 路由现在已经安装上了

```

R2#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
Known via "ospf 1", distance 130, metric 74, type intra area
Redistributing via ospf 1
Last update from 131.108.1.1 on Serial0, 01:39:09 ago
Routing Descriptor Blocks:
* 131.108.1.1, from 131.108.2.1, 14:39:09 ago, via Serial0
Route metric is 64, traffic share count is 1

```

2. OSPF 没有在路由表中安装任何路由——原因: IP 地址在通过双重串行链路连接的路由器上配反了

在双重串行接口上的 IP 地址可能引起这个问题。OSPF 形成一个 FULL 邻接体关系, 因为邻居仍可达, 但是这导致了 OSPF 数据库中的不一致。

当在某一端分配了错误的 IP 子网或掩码时也会发生这个问题。这也会在 OSPF 数据库中形成不一致。

图 9-68 显示了一个网络设置, 其中在串行接口上的地址配反了。例如, R1 的 Serial 0 地址为 131.108.1.1/24。这个接口的另一端进入 R2 的 Serial 0 中, 它在 Serial 0 下定义的地址为 131.108.2.1/24。相似的事情可以观察到发生在 Serial 1 接口上。这显然看起来是地址配反了。

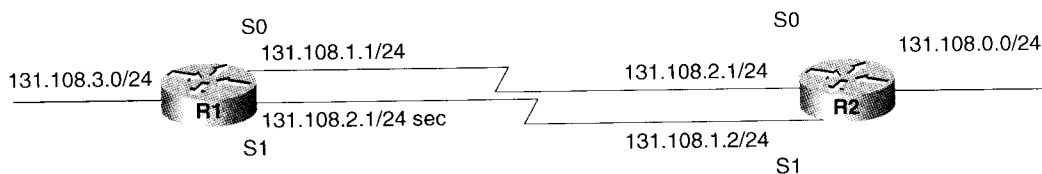


图 9-68 OSPF 网络, 其中路由器串行接口 IP 地址配反了

例 9-190 显示了由于数据库中的不一致, R2 没有在路由表中安装任何路由。

例 9-190 R2 在它的路由表中没有安装任何路由

```

R2#show ip route ospf
R2#

```

(待续)

图 9-69 显示了解决这个问题应遵循的流程图。

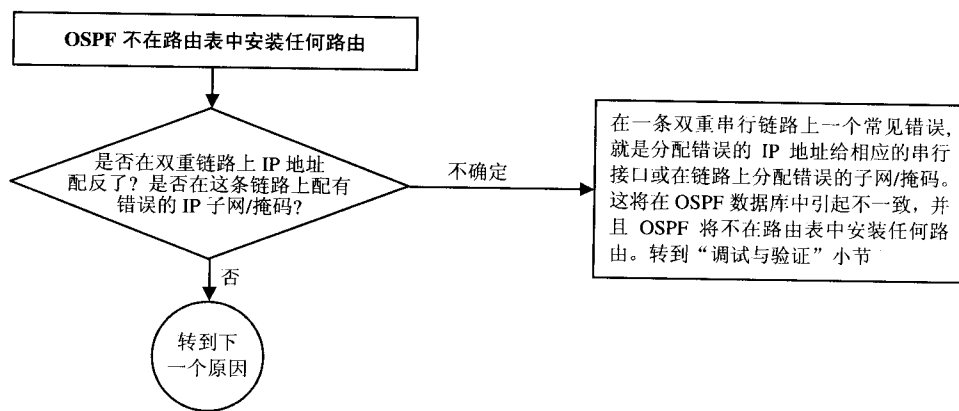


图 9-69 解决问题流程图

(1) 调试与验证

例 9-191 显示了 **show ip ospf neighbor** 的输出, 表明 OSPF 在两条串行链路上都形成了 FULL 邻接体关系。地址行显示邻居地址和接口不匹配。例如, 在例 9-191 中, R2 有两个邻居。因为 R2 的 Serial 0 地址在范围 131.108.2.0/24 中, 如图 9-68 所示, 邻居的 Serial 0 的地址也应该在同一范围内, 但它显示 131.108.1.1 为 Serial 0 上的一个邻居。

例 9-191 **show ip ospf neighbor** 命令输出表示在 R1 的串行链路上形成了 OSPF 邻接体关系

R2#show ip ospf neighbor					
Neighbor ID	Pri	State	Dead Time	Address	Interface
131.108.2.1	1	FULL/ -	00:00:37	131.108.1.1	Serial0
131.108.2.1	1	FULL/ -	00:00:31	131.108.2.1	Serial1

(2) 解决方法

要解决这个问题, 需在正确的相应接口上分配 IP 地址。要么在 R1 的串行链路上改变它的 IP 地址, 要么在 R2 的串行链路上改变 R2 的 IP 地址。在这个例子中, R2 的串行链路已被改变来匹配 R1 的地址。Serial 0 的 IP 地址和 Serial 1 的 IP 地址已经被交换了, 如例 9-192 所示。

例 9-192 在 R2 的串行链路上改变 IP 地址

```

R2#
interface serial 0
no ip address
ip address 131.108.1.2 255.255.255.0
!
interface serial 1
no ip address
ip address 131.108.2.1 255.255.255.0
  
```

例 9-193 显示了在解决这个问题之后, R2 在它的路由表中安装了 OSPF 路由。

例 9-193 R2 的路由表表明问题已经解决了

```
R2#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
  Known via "ospf 1", distance 130, metric 74, type intra area
  Redistributing via ospf 1
  Last update from 131.108.1.1 on Serial0, 01:39:09 ago
  Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.2.1, 14:39:09 ago, via Serial0
      Route metric is 64, traffic share count is 1

R2#
```

3. OSPF 没有在路由表中安装任何路由——原因: 一边是有编号的而另一边是无编号的点到点链路

当 OSPF 为点到点链路创建一个路由器 LSA 时, 它坚持如下原则来将链路 ID 字段和链路数据字段填入到路由器 LSA 中 (见表 9-1)。

表 9-1 OSPF 点到点有编号和无编号链路的链路 ID 和链路数据字段值

类 型	描 述	链 路 ID	链 路 数 据
1	点到点有编号	邻居的路由器 ID	接口 IP 地址
1	点到点无编号	邻居的路由器 ID	MIB-II ifIndex 值

表 9-1 显示了无编号点到点链路的链路数据字段应该有一个 MIB-II ifIndex 值。因为有编号接口的链路数据值和无编号接口的数据链路值不匹配, 这就导致了 OSPF 数据库中的不一致。

图 9-70 显示了一个网络设置, 其中 R1 的串行链路是无编号的回环接口, 而 R2 的串行链路是有编号的。



图 9-70 OSPF 网络有一条串行链路, 这条串行链路是一个无编号回环接口

图 9-71 显示了解决这个问题应遵循的流程图。

(1) 调试与验证

例 9-194 显示了 OSPF 数据库中的不一致。R1 的路由器 LSA 显示了 Serial 0 的链路数据字段中的 MIB-II IfIndex 值, 而 R2 的路由器 LSA 在链路数据字段中显示了路由器的串行接口地址。

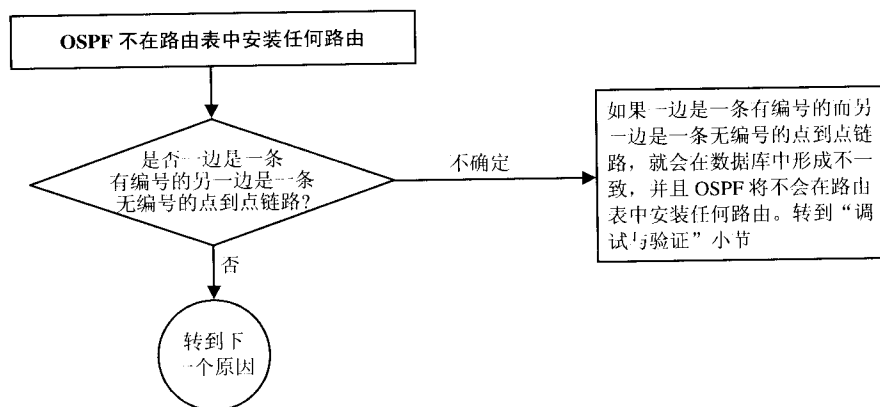


图 9-71 解决问题流程图

例 9-194 检查 OSPF 数据库的不一致问题

```
R2#show ip ospf database router
```

```
OSPF Router with ID (131.108.1.2) (Process ID 1)
```

```
Router Link States (Area 0)
```

```
Adv Router is not-reachable
```

```
LS age: 855
```

```
Options: (No TOS-capability, DC)
```

```
LS Type: Router Links
```

```
Link State ID: 131.108.1.1
```

```
Advertising Router: 131.108.1.1
```

```
LS Seq Number: 8000000D
```

```
Checksum: 0x55AD
```

```
Length: 60
```

```
Number of Links: 1
```

```
Link connected to: another Router (point-to-point)
```

```
(Link ID) Neighboring Router ID: 131.108.1.2
```

```
(Link Data) Router Interface address: 0.0.0.4
```

```
Number of TOS metrics: 0
```

```
TOS 0 Metrics: 64
```

```
R1#show ip ospf database router
```

```
OSPF Router with ID (131.108.1.1) (Process ID 1)
```

```
Router Link States (Area 0)
```

```
Adv Router is not-reachable
```

```
LS age: 855
```

```
Options: (No TOS-capability, DC)
```

```
LS Type: Router Links
```

```
Link State ID: 131.108.1.2
```

(待续)

```
Advertising Router: 131.108.1.2
LS Seq Number: 8000000D
Checksum: 0x55AD
Length: 60
Number of Links: 1

Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 131.108.1.1
(Link Data) Router Interface address: 131.108.1.2
Number of TOS metrics: 0
TOS 0 Metrics: 64
```

例 9-195 显示了在 R1 和 R2 上的配置，表明 R1 的串行接口是一个无编号的回环地址，而 R2 的串行接口是有编号的。

例 9-195 R1 和 R2 的串行接口配置

```
R1#
interface Loopback0
 ip address 131.108.1.1 255.255.255.0
!
interface Serial0
 ip unnumbered Loopback0
!
router ospf 1
 network 131.108.0.0 0.0.255.255 area 0

R2#
interface Serial0
 ip address 131.108.1.2 255.255.255.0
!
router ospf 1
 network 131.108.0.0 0.0.255.255 area 0
```

(2) 解决方法

要解决这个问题，需确信两边要么都是有编号的点到点链路要么都是无编号的点到点链路。例 9-196 显示了新的配置以解决这个。在这个例子中，R1 的串行接口分配了一个 IP 地址。

例 9-196 在 R1 的串行接口分配一个 IP 地址，这个串行接口以前是无编号的

```
R1#
interface Serial0
 ip address 131.108.1.1 255.255.255.0
!
router ospf 1
 network 131.108.0.0 0.0.255.255 area 0

R2#
interface Serial0
 ip address 131.108.1.2 255.255.255.0
!
router ospf 1
 network 131.108.0.0 0.0.255.255 area 0
```

(待续)

例 9-197 显示了在解决这个问题后, R2 在路由表中安装路由了。

例 9-197 确认 OSPF 路由安装

```
R2#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
  Known via "ospf 1", distance 130, metric 74, type intra area
  Redistributing via ospf 1
  Last update from 131.108.1.1 on Serial0, 01:39:09 ago
  Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.2.1, 14:39:09 ago, via Serial0
      Route metric is 64, traffic share count is 1
R2#
```

4. OSPF 不在路由表中安装任何路由——原因: 分发表阻止安装路由

OSPF 是一种链路状态协议, 当它与邻居形成邻接体的时候, 它与那个邻居同步它们的数据库。因为它的链路状态性质, 在 OSPF 中过滤是不可能的。

配置 `distribute-list in` 阻止 OSPF 路由安装到路由表中。这并不表示路由从 OSPF 数据库消失了。只是简单地表示在 OSPF 安装路由到路由表中以前, 它检查分发表并只安装那些分发表中允许的网络。无论如何, 它还是将路由保持在数据库中。

图 9-72 显示了一个网络设置, 其中 OSPF 不在路由表中安装任何路由。特别地, 路由器 2 没有在它的路由表中看到任何 OSPF 路由。



图 9-72 导致路由安装问题的 OSPF 网络

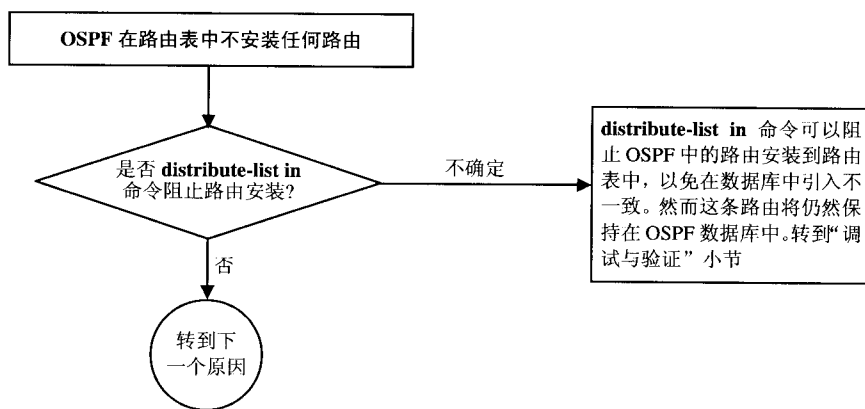


图 9-73 解决问题流程图

(1) 调试与验证

例 9-198 显示了在 R2 上的配置, 表明 R2 配置了 `distribute-list in` 命令, 因而在路由表中没有安装任何 OSPF 路由。用于分发表中的 `access-list 1` 仅仅允许 10/8 和 20/8 网络安装到路由表中。剩余的网络明确地被拒绝。例 9-198 也显示了网络 131.108.3.0 不在路由表中, 因

为它被分发表拒绝了。

例 9-198 R2 配置了一个分发表

```
R2#
!
router ospf 1
 network 131.108.0.0 0.0.255.255 area 0
 distribute-list 1 in
!
access-list 1 permit 10.0.0.0
access-list 1 permit 20.0.0.0

R2#show ip route 131.108.3.0
%Network not in table
R2#
```

(2) 解决方法

如果一个网络在数据库中，但是它不在路由表中，确信这个网络在分发表中是允许的。

例 9-199 显示了解决这个问题的新配置。在这个配置中，网络 131.108.3.0 是允许的。

例 9-199 配置 R2 的分发表来允许网络 131.108.3.0/24

```
R2#
!
router ospf 1
 network 131.108.0.0 0.0.255.255 area 0
 distribute-list 1 in
!
access-list 1 permit 10.0.0.0
access-list 1 permit 20.0.0.0
access-list 1 permit 131.108.3.0 0.0.0.255
```

例 9-200 显示了在解决这个问题后网络 131.108.3.0 / 24 出现在了路由表中。

例 9-200 确认路由 131.108.3.0 安装了

```
R2#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
  Known via "ospf 1", distance 130, metric 74, type intra area
  Redistributing via ospf 1
  Last update from 131.108.1.1 on Serial0, 01:09:19 ago
  Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.2.1, 14:39:09 ago, via Serial0
      Route metric is 64, traffic share count is 1

R2#
```

5. OSPF 不在路由表中安装任何路由——原因：在一个广播网络类型的完全网状互连帧中继网络中有中断的 PVC

OSPF 广播网络类型将永远不会定义在一个非完全网状互连介质上。有时，介质是完全

网状互连的，但是第 2 层连接不稳定。在那种情况下，当第 2 层出故障了，就会产生不一致并且介质会再次变为非完全网状互连。

图 9-74 显示了一个遭受这个问题的网络。在 R1 和 R2 之间的 PVC 断开之前，R2 是一个 DR 而 R3 是一个 BDR。

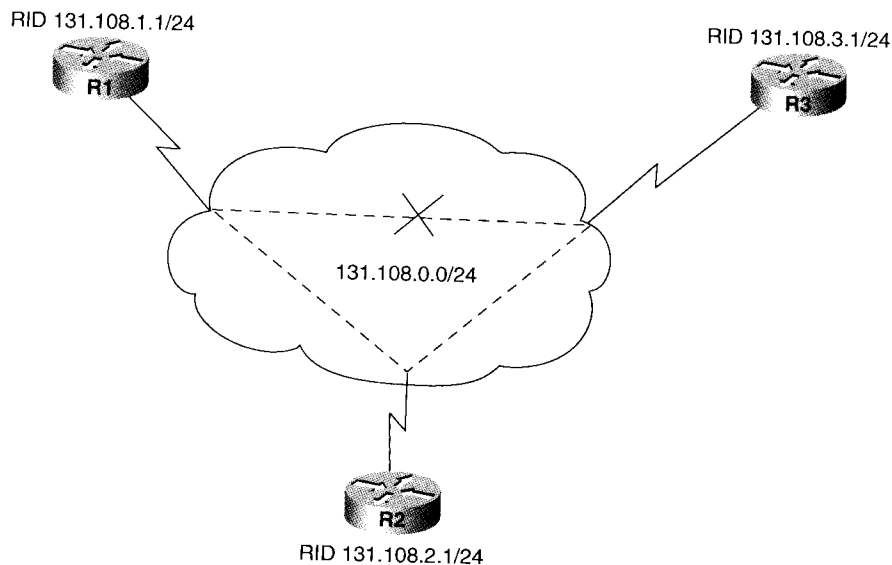


图 9-74 OSPF 网络中的一个中断的 PVC 引起了这个问题，这个 PVC 处在一个有着广播网络类型的完全网状互连帧中继网络中

例 9-201 显示了 R1 不在路由表中安装任何路由。

例 9-201 R1 不安装任何路由

```
R1#show ip route ospf
R1#
```

图 9-75 显示了解决这个问题应遵循的流程图。

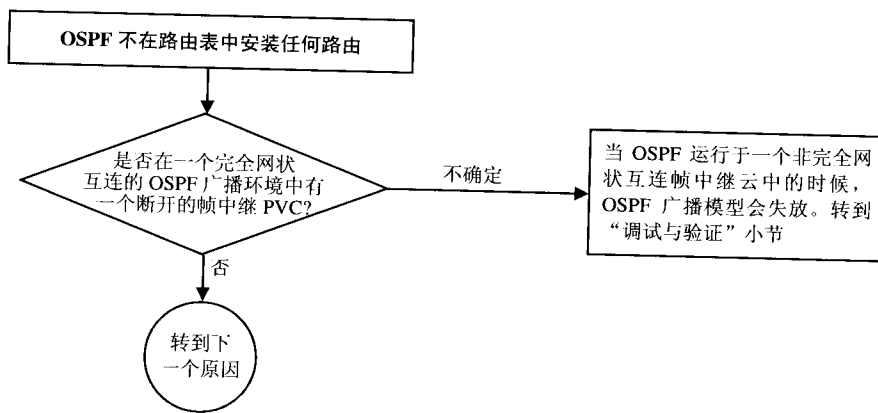


图 9-75 解决问题流程图

(1) 调试与验证

例 9-202 显示了 R1、R2 和 R3 上的配置。配置显示在帧中继云中所有路由器上网络类型为广播类型。

例 9-202 在 R1、R2 和 R3 上确认网络类型

```

R1#
!
interface Serial0.1 multipoint
 ip address 131.108.0.1 255.255.255.0
 ip ospf network broadcast

R2#
!
interface Serial0.1 multipoint
 ip address 131.108.0.2 255.255.255.0
 ip ospf network broadcast

R3#
!
interface Serial0.1 multipoint
 ip address 131.108.0.3 255.255.255.0
 ip ospf network broadcast

```

例 9-203 显示了在所有这 3 个路由器上 **show ip ospf neighbor** 的输出。在 R1 上的输出显示它认为 R2 是 DR。可是，实际的 DR 是 R3，如图 9-74 所示，因为它有着最大的路由器 ID。因为在 R1 和 R3 之间的链路断开了，所以 R1 宣布 R2 是 DR。

例 9-203 确定指定路由器 (Designated Router)

```

R1#show ip ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
131.108.2.1 1 FULL/DR 00:00:31 131.108.0.2 Serial0.1

R2#show ip ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
131.108.1.1 1 FULL/DROTHER 00:00:34 131.108.0.1 Serial0.1
131.108.3.1 1 FULL/DR 00:00:33 131.108.0.3 Serial0.1

R3#show ip ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
131.108.2.1 1 FULL/BDR 00:00:31 131.108.0.2 Serial0.1

```

例 9-204 显示了在 R1 上的路由器 LSA 输出显示了发往 R1、R2 和 R3 的路由器 LSA。R1 仍然认为 DR 是 R2 而不是 R3。R1 显示帧中继链路是一个端链路，因为它不能在数据库中发现任何由 R2 产生的网络 LSA。这条帧中继链路在 R2 和 R3 的数据库中都定义为一条中继链路。这在 OSPF 数据库中产生了一致。

例 9-204 R1、R2 和 R3 的数据库显示存在不一致

```

R1#show ip ospf database router

OSPF Router with ID (131.108.1.1) (Process ID 1)

```

(待续)

Router Link States (Area 0)

LS age: 148
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.1.1
Advertising Router: 131.108.1.1
LS Seq Number: 8000000B
Checksum: 0x55A
Length: 48
Number of Links: 2

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.0.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 64

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.1.1
(Link Data) Network Mask: 255.255.255.255
Number of TOS metrics: 0
TOS 0 Metrics: 1

Adv Router is not-reachable
LS age: 1081
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 131.108.2.1
Advertising Router: 131.108.2.1
LS Seq Number: 80000006
Checksum: 0x4F72
Length: 48
Number of Links: 2

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.2.1
(Link Data) Network Mask: 255.255.255.255
Number of TOS metrics: 0
TOS 0 Metrics: 1

Link connected to: a Transit Network
(Link ID) Designated Router address: 131.108.0.3
(Link Data) Router Interface address: 131.108.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 64

Adv Router is not-reachable
LS age: 306
Options: (No TOS-capability, DC)

(待续)

```

LS Type: Router Links
Link State ID: 131.108.3.1
Advertising Router: 131.108.3.1
LS Seq Number: 80000007
Checksum: 0xC185
Length: 48
Number of Links: 2

Link connected to: a Stub Network
(Link ID) Network/subnet number: 131.108.3.1
(Link Data) Network Mask: 255.255.255.255
Number of TOS metrics: 0
TOS 0 Metrics: 1

Link connected to: a Transit Network
(Link ID) Designated Router address: 131.108.0.3
(Link Data) Router Interface address: 131.108.0.3
Number of TOS metrics: 0
TOS 0 Metrics: 64

```

(2) 解决方法

在这种情况下的解决方法是运行一个点到多点网络类型。配置点到多点网络类型后，由于没有 DR / BDR 选举，泛洪增加了。可是如果有任何连接问题，它将不会创建任何黑洞(black holes)。

因此，在可靠性和增加泛洪之间存在权衡。在一个不稳定的帧中继链路上选择点到多点网络类型提供了可靠性，而如果选择广播网络类型，那么每当任何第 2 层不稳定发生时将会产生不一致性。

图 9-205 显示了在 R1、R2 和 R3 上新的配置改变。网络类型现在是点到多点类型了。

例 9-205 为 R1、R2 和 R3 配置点到多点接口

```

R1#
!
interface Serial0.1 multipoint
 ip address 131.108.0.1 255.255.255.0
 ip ospf network point-to-multipoint

R2#
!
interface Serial0.1 multipoint
 ip address 131.108.0.2 255.255.255.0
 ip ospf network point-to-multipoint

R3#
!
interface Serial0.1 multipoint
 ip address 131.108.0.3 255.255.255.0
 ip ospf network point-to-multipoint

```

例 9-206 显示了在配置改变之后，R1 开始从 R2 接收 OSPF 路由。

(待续)

例 9-206 确认 R1 从 R2 接收 OSPF 路由

```

R1#show ip route 131.108.3.0
Routing entry for 131.108.3.0/24
  Known via "ospf 1", distance 130, metric 129, type intra area
  Redistributing via ospf 1
  Last update from 131.108.0.2 on Serial0.1, 00:00:19 ago
  Routing Descriptor Blocks:
    * 131.108.0.2, from 131.108.3.1, 14:39:09 ago, via Serial0.1
      Route metric is 64, traffic share count is 1
R1#

```

9.4.2 问题：OSPF 不在路由表中安装外部路由

当 OSPF 重发布任何路由时，不管它们来自于直接连接的、静态的或者来自于一个不同的路由协议，它都为这些外部路由产生一个类型 5 LSA。这些类型 5 LSA 被泛洪到每个 OSPF 路由器中，只有那些在端区或者 NSSA 区域中的路由器是例外。有时，问题是外部路由在 OSPF 数据库中，但是它们没有被安装到路由表中。

这个问题最常见的原因如下：

- 不能通过区域内或区域间路由了解到转发地址。
- ABR 不产生类型 4 汇总 LSA。

1. OSPF 不在路由表中安装外部路由——原因：不能通过区域内或区域间路由了解到转发地址

当 OSPF 了解到一个外部 LSA 时，它确信在它安装这条路由到路由表之前，要确实通过 OSPF 区域内或者区域间路由了解到转发地址。如果没有通过区域内或区域间路由了解到转发地址，那么 OSPF 将不在路由表中安装路由。这与 RFC 2328 标准一致。

图 9-76 显示了有以下详细规定的一个网络：

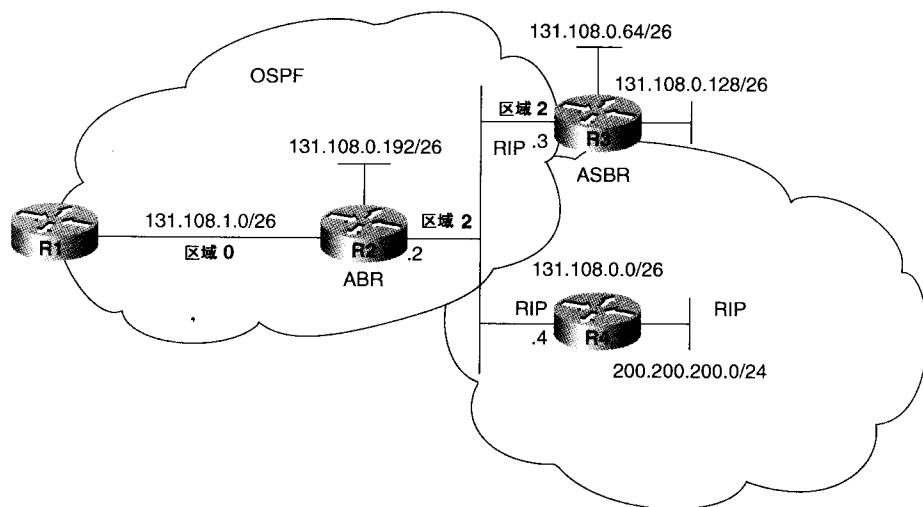


图 9-76 OSPF 网络中存在一个不能在路由表中安装外部路由的问题

- R3 是一个 ASBR，它重发布 RIP 路由到 OSPF 中。
- R4 与 R3 之间运行 RIP。
- R4 通过 RIP 了解到转发地址 200.200.200.0/24。
- R2 与 R3 之间运行 OSPF。
- R2 是 ABR。

例 9-207 显示了为 200.200.200.0 运行 **show ip route** 的输出。这个网络属于一个 RIP 域。因为 RIP 在 R3 上被重发布进入 OSPF 中，所以所有的 OSPF 路由器将视这条路由为 OSPF 外部路由。可是，R1 没有在它的路由表中看到这条路由。

例 9-207 R1 在它的路由表中缺少 RIP 路由 200.200.200.0

```
R1#show ip route 200.200.200.0
% Network not in table
R1#
```

图 9-77 显示了解决这个问题应遵循的流程图。

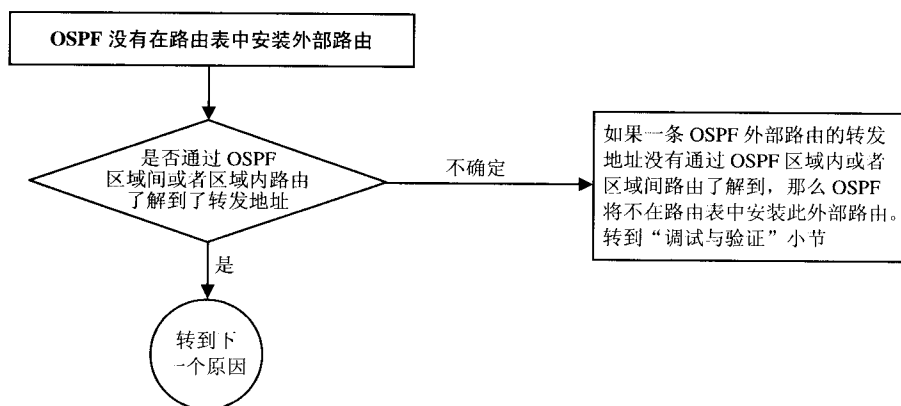


图 9-77 解决问题流程图

(1) 调试与验证

例 9-208 显示了为路由 200.200.200.0 产生的外部 LSA。这个外部 LSA 存在于 OSPF 数据库中，但是路由仍然没有被安装到路由表中。这也有一个转发地址包括在了这个外部 LSA 中。

例 9-208 用于 RIP 路由 200.200.200.0 的外部 LSA 数据库

```
R1#show ip ospf database external 200.200.200.0

OSPF Router with ID (131.108.1.1) (Process ID 1)

Type-5 AS External Link States

LS age: 14
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 200.200.200.0 (External Network Number )
```

(待续)

```

Advertising Router: 131.108.0.129
LS Seq Number: 80000001
Checksum: 0x88BE
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 20
    Forward Address: 131.108.0.4
    External Route Tag: 0

```

例 9-209 显示了通过 OSPF 外部路由了解到的发往转发地址的路由。

例 9-209 发往转发地址的路由，这个路由是通过 OSPF 外部路由了解到的

```

R1#show ip route 131.108.0.4
Routing entry for 131.108.0.0/26
    Known via "ospf 1", distance 110, metric 20, type extern 2, forward metric 70
    Redistributing via ospf 1
    Last update from 131.108.1.2 on Serial0, 00: 00: 40 ago
    Routing Descriptor Blocks:
    * 131.108.1.2, from 131.108.0.129, 00: 00: 40 ago, via Serial0
      Route metric is 20, traffic share count is 1

```

例 9-210 显示了 ABR 用一个 **area range** 命令汇总路由 131.108.0.0/24，因此更详细的区域内路由被汇总进了这条路由中。这个范围汇总了在 131.108.0.0/16 范围中的所有路由。

例 9-210 R2 汇总区域内路由为 131.108.0.0/24

```

R2#
router ospf 1
    network 131.108.1.0 0.0.0.255 area 0
    network 131.108.0.0 0.0.0.255 area 2
    area 2 range 131.108.0.0 255.255.255.0

```

例 9-211 显示了 ASBR 在处理从 RIP 到 OSPF 的重发布。它也显示了在 131.108.0.0/16 范围内直接连接的网络被重发布进 OSPF 中，因为 RIP 拥有这些直连路由。这个配置重发布 131.108.0.4/26，它是一个 R3 上的直连端口。这个子网覆盖了在例 9-209 中出现的转发地址。

例 9-211 R3 重分布在 131.108.0.0/16 范围内的 RIP 路由到 OSPF 中

```

R3#
router ospf 1
    redistribute rip subnets
    network 0.0.0.0 255.255.255.255 area 2
!
router rip
    network 131.108.0.0

```

(2) 解决方法

R1 看到了通过 OSPF 外部 LSA 了解到的转发地址，因为 R3 在 **router ospf** 下连接有

redistributer 语句。这为 R3 的直连接口引入了一条更明确的路由。这也包含了转发地址子网，即 131.108.0.4/26。为这个子网产生的区域内路由也被 R2 禁止，因为 R2 汇总 131.108.0.0/16。因为更详细的路由总是首选的，所以 R1 宁可使用 131.108.0.4 / 26 这条更详细的外部路由而不是使用 131.108.0.0/16 这条不详细的汇总路由。

这个问题可以有两种解决方式：

- 不在 ABR 上进行汇总。
- 在 ASBR 上过滤掉被重发布进 OSPF 的直连子网。

要实现第一种解决方案，可转到 ABR 并去掉 **area range** 命令。

例 9-212 显示了 ABR 上的新配置以解决这个问题。

例 9-212 在 ABR 上去掉汇总功能

```
R2#
router ospf 1
network 131.108.1.0 0.0.0.255 area 0
network 131.108.0.0 0.0.0.255 area 2
no area 2 range 131.108.0.0 255.255.255.0
```

要实现第二种解决方案，可转到 ASBR 并增加一个过滤器来控制重发布路由。

例 9-213 显示了解决这个问题的新配置。过滤器现在阻止路由 131.108.0.0/26 作为一条外部路由由重发布到 OSPF 中，因为仅有 200.200.200.0 是允许的，所有其他的路由都将被拒绝。

例 9-213 配置 ASBR 来过滤直连的路由

```
R3#
router ospf 1
redistribute rip subnets route-map no_connected
network 0.0.0.0 255.255.255.255 area 2
!
router rip
network 131.108.0.0
!
route-map no_connected permit 10
match ip address 1
!
access-list 1 permit 200.200.200.0 0.0.0.255
```

例 9-214 显示了在解决这个问题后，R1 开始在路由表中安装路由了，因为现在通过 OSPF 区域间路由而了解到转发地址了。

例 9-214 确认在 R1 上再次安装外部路由

```
R1#show ip route 200.200.200.0
Routing entry for 200.200.200.0/24
Known via "ospf 2", distance 110, metric 20, type extern 2, forward metric 128
Redistributing via ospf 2
Last update from 131.108.1.2 on Serial0.1, 00:47:24 ago
```

(待续)

Routing Descriptor Blocks:

```
* 131.108.1.2, from 131.108.0.29, 00:47:24 ago, via Serial0.1
  Route metric is 20, traffic share count is 1
```

例 9-215 也显示了现在通过 OSPF 区域间而不是 OSPF 外部路由了解到转发地址了。

例 9-215 现在通过 OSPF 区域间路由了解到转发地址了

```
R1#show ip route 131.108.0.4
```

```
Routing entry for 131.108.0.4/26
```

```
Known via "ospf 2", distance 110, metric 64, type inter area
```

```
Redistributing via ospf 2
```

```
Last update from 131.108.1.2 on Serial0.1, 00:50:25 ago
```

```
Routing Descriptor Blocks:
```

```
* 131.108.1.2, from 131.108.0.193, 00:50:25 ago, via Serial0.1
  Route metric is 64, traffic share count is 1
```

2. OSPF 不在路由表中安装外部路由——原因：ABR 不产生类型 4 汇总 LSA

类型 4 汇总 LSA 的一个功能就是宣布 ASBR 到其他区域的可达性。如果 ASBR 存在于同一个区域中，这个类型 4 LSA 也就不需要了。

如果 ASBR 不连接到区域 0，那么它不产生类型 4 汇总 LSA。为了产生一个类型 3 或者类型 4 的汇总 LSA，路由器必须有一条到区域 0 的连接。因此，外部路由将不被安装于网络中。第 8 章非常详细地讲述了类型 3 和类型 4 LSA。

图 9-78 显示了一个网络，其中 R3 重发布 RIP 路由到 OSPF 中。

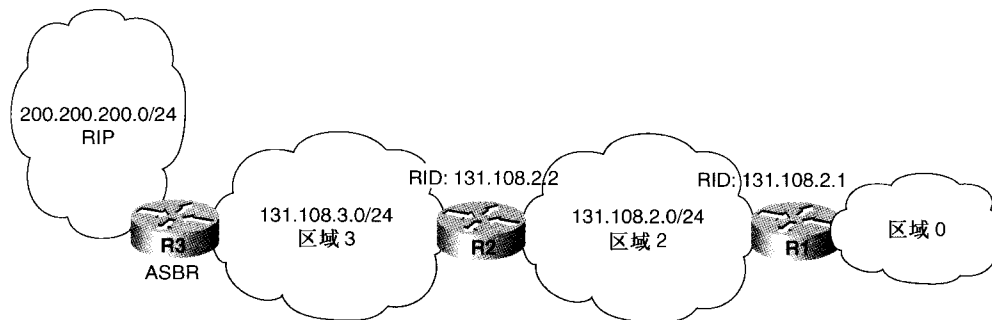


图 9-78 因为缺少类型 4 汇总 LSA，OSPF 网络存在外部路由安装问题

例 9-216 显示了 R1 没有安装外部路由 200.200.200.0 到路由表中。

例 9-216 R1 没有安装外部路由 200.200.200.0

```
R1#show ip route 200.200.200.0
```

```
% Network not in table
```

```
R1#
```

图 9-79 显示了解决这个问题应遵循的流程图。

(1) 调试与验证

例 9-217 显示了存在于外部数据库中的路由。

ABR
输出
路由

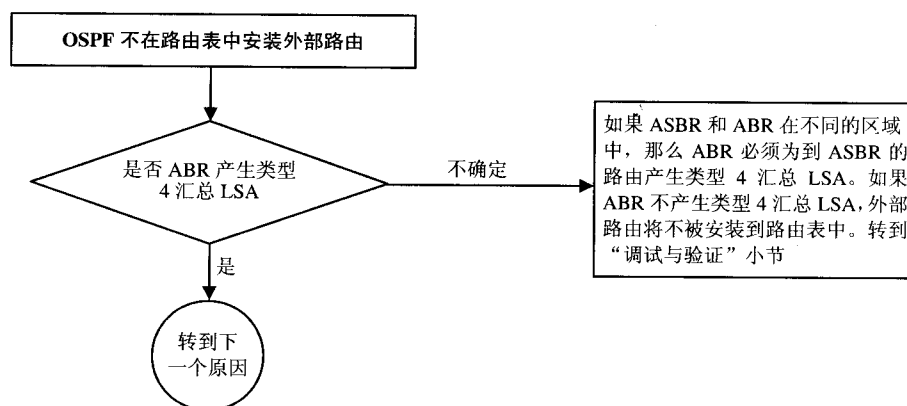


图 9-79 解决问题流程图

例 9-217 确认路由 200.200.200.0 存在于外部 OSPF 数据库中

```

R1#show ip ospf database external 200.200.200.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

Type-5 AS External Link States

LS age: 199
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 200.200.200.0 (External Network Number)
Advertising Router: 131.108.3.3
LS Seq Number: 80000001
Checksum: 0x4B3A
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 20
    Forward Address: 0.0.0.0
    External Route Tag: 0
  
```

例 9-218 显示了没有为这条路由产生类型 4 LSA。

例 9-218 对外部路由没有类型 4 LSA 存在

```

R1#show ip ospf database asbr-summary 131.108.3.3

OSPF Router with ID (131.108.2.1) (Process ID 1)
  
```

下一个逻辑步骤是到 ABR 上看它是否真是一个 ABR。如果它不认为它自己是一个 ABR，它将不产生任何类型 3 或者类型 4 汇总 LSA。例 9-219 显示了在 `show ip ospf` 的输出中 R2 不标识它自己是 ABR。如果 R2 是 ABR，那么输出将显示“这是一个区域边界路由器”。

例 9-219 R2 不认为它自己是 ABR

```
R2#show ip ospf
Routing Process "ospf 1" with ID 131.108.2.2
Supports only single TOS(TOS0) routes
SPF schedule delay 5 secs, Hold time between two SPF's 10 secs
```

(2) 解决方法

在这个例子中, R2 不产生类型 4 汇总 LSA, 因为它不与区域 0 相连。要产生类型 3 或者类型 4 汇总 LSA, 路由器必须有一条到区域 0 的连接。

要解决这个问题, 需通过物理连接或者通过创建一条虚链路的虚连接来将 R2 与区域 0 相连, 如例 9-220 所示。要了解关于虚链路的更多信息, 请参阅第 8 章。

例 9-220 在 R2 和 R1 之间配置虚链路

```
R1#
router ospf 1
area 2 virtual-link 131.108.2.2

R2#
router ospf 1
area 2 virtual-link 131.108.2.1
```

通过在 R2 上配置一条虚链路, 该路由器现在与区域 0 虚拟连接了; 因此, 它现在认为它自己是 ABR 了。例 9-221 显示了在 R2 连接到区域 0 之后, **show ip ospf** 的输出表明它是 ABR 了。将这个输出与例 9-219 中的输出相比较, 在例 9-219 中路由器不认为它自己是 ABR。

例 9-221 确认 R1 现在注意到了它的 ABR 状态

```
R2#show ip ospf
Routing Process "ospf 1" with ID 131.108.2.2
Supports only single TOS(TOS0) routes
It is an area border router
SPF schedule delay 5 secs, Hold time between two SPF's 10 secs
```

现在, 再次到 R1 上看是否收到了类型 4 LSA。例 9-222 显示在 R2 上配置改变之后, 它已经开始产生类型 4 汇总 LSA 到区域 3 中了。

例 9-222 R2 现在产生类型 4 LSA 到区域 2 中, 并且 R1 收到它了

```
R1#show ip ospf database asbr-summary

OSPF Router with ID (131.108.2.1) (Process ID 1)

Summary ASB Link States (Area 2)

LS age: 17
Options: (No TOS-capability, DC)
LS Type: Summary Links(AS Boundary Router)
```

(待续)

```

Link State ID: 131.108.3.3 (AS Boundary Router address)
Advertising Router: 131.108.2.2
LS Seq Number: 80000001
Checksum: 0xE269
Length: 28
Network Mask: /0
TOS: 0 Metric: 64

```

因为现在类型 4 LSA 已经产生了, 所以 R1 在它的路由表中安装了外部路由。例 9-223 显示了在解决这个问题之后, 外部路由 200.200.200.0 / 24 已经在 R1 的路由表中了。

例 9-223 外部路由 200.200.200.0 现在已经在 R1 的路由表中了

```

R1#show ip route 200.200.200.0
Routing entry for 200.200.200.0/24
  Known via "ospf 2", distance 110, metric 20, type extern 2, forward metric 128
  Redistributing via ospf 2
  Last update from 131.108.2.2 on Serial0.1, 00:47:24 ago
  Routing Descriptor Blocks:
    * 131.108.2.2, from 131.108.3.3, 00:47:24 ago, via Serial0.1
      Route metric is 20, traffic share count is 1

```

9.5 OSPF 重发布问题疑难解析

本节讨论与 OSPF 路由重发布相关的问题。当 OSPF 中一个路由器做路由重发布工作的时候, 它就变成了一个 ASBR。重发布进 OSPF 的路由可以是直接相连路由、静态路由或者是动态地从其他路由协议或者其他的 OSPF 进程了解到的路由。

下面是在重发布中会发生的问题:

- ASBR 不通告被重发布的路由。
- OSPF 不在路由表中安装外部路由。

第二个问题在早先关于 OSPF 路由安装问题的小节中已经讨论过了。第一个问题在接下来这一节中讨论。

问题: OSPF 邻居不通告外部路由

当一条路由已经知道是一条直连的或静态的, 或者当任何其他路由协议重发布进 OSPF 的时候, 就会为这条路由产生一个外部 LSA。如果 OSPF 路由器甚至在路由重发布之后还不通告外部路由, 这表示在处理路由重发布的路由器上存在一个问题。大部分情况下, 问题的根源是配置错误。

这个问题最常见的可能原因如下:

- 在 ASBR 配置中缺少关键字 **subnets**。
- **distribute-list out** 命令阻止路由。

图 9-80 显示了正面临这个问题的一个网络。在该图中, R1 在以太网上运行 RIP 并且重

要产生类型 3 或

将 R2 与区域 0

因此, 它现在认为
的输出表明它是
为它自己是 ABR。

配置改变之后, 它

(待续)

发布 RIP 路由到 OSPF 中。

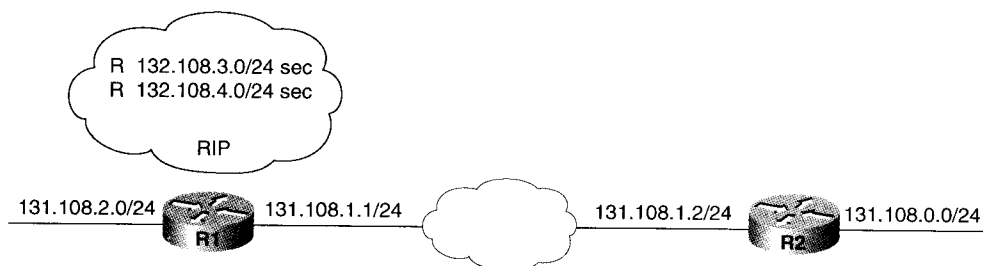


图 9-80 网络设置显示在 OSPF 中的重发布

1. OSPF 邻居不通告外部路由——原因：在 ASBR 的配置中缺少关键字 subnets

当任何其他路由协议被重发布进 OSPF 中的时候，如果被重发布的网络是子网，那么你必须要在 OSPF 配置下定义关键字 subnets。如果没有加上 subnets 关键字，OSPF 将在产生外部 LSA 的时候忽略所有的子网路由。

在直连的或者静态的路由被重发布进入 OSPF 或者从 OSPF 中输出的时候，这种情况就会出现。在这种情况下，应用同一条规则：必须键入 subnets 关键字来重发布子网路由。

图 9-81 显示了解决这个问题应遵循的流程图。

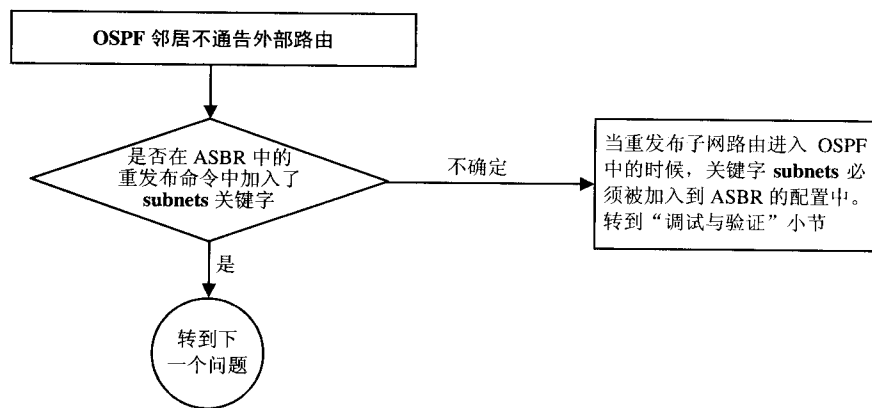


图 9-81 解决问题流程图

(1) 调试与验证

例 9-224 显示了为 132.108.3.0 运行命令 `show ip ospf database external` 的输出。输出显示没有 LSA 信息，这表示 R1 没有为 132.108.3.0 产生外部 LSA。

例 9-224 R1 没有为 132.108.3.0 产生一个外部 LSA

```
R1#show ip ospf database external 132.108.3.0

OSPF Router with ID (131.108.2.1) (Process ID 1)

R1#
```

例 9-225 显示了在 R1 的 OSPF 配置。配置显示 RIP 的重发布缺少 subnets 关键字。

例 9-225 R1 的配置缺少为 RIP 重发布而使用的 subnets 关键字

```

R1#
router ospf 1
  redistribute rip
  network 131.108.1.0 0.0.0.255 area 0
  network 131.108.2.0 0.0.0.255 area 0
  !
router rip
  network 132.108.0.0
  !

```

(2) 解决方法

要解决这个问题，需增加 **subnets** 关键字到 **redistribution** 命令中。例 9-226 显示了解决这个问题的正确配置。

例 9-226 在 R1 的配置中增加 subnets 关键字

```

R1#
router ospf 1
  redistribute rip subnets
  network 131.108.1.0 0.0.0.255 area 0
  network 131.108.2.0 0.0.0.255 area 0

```

在增加了 **subnets** 关键字之后，OSPF 重发布所有的子网路由，例如，131.108.3.0 是一条掩码为/24 的子网路由。例 9-227 显示了 R1 开始为 132.108.3.0 和 132.108.4.0 产生外部 LSA。

例 9-227 确认 R1 为 132.108.3.0 和 132.108.4.0 产生外部路由

```

R1#show ip ospf database external 132.108.3.0

      OSPF Router with ID (131.108.2.1) (Process ID 1)

Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.3.0 (External Network Number )
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 1

```

(待续)

```

R1#
R1#show ip ospf database external 132.108.4.0

      OSPF Router with ID (131.108.2.1) (Process ID 1)

Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.4.0 (External Network Number)
Advertising Router: 131.108.2.1
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 1

R1#

```

2. OSPF 邻居不通告外部路由——原因: distribute-list out 命令阻止路由

当在 ASBR 上配置了 `distribute-list out` 命令的时候,与分发表相关联的访问列表被检查并且为在分发表中明确允许的网络产生类型 5 外部 LSA。所有其他的网络都被拒绝,并且没有为这些网络产生类型 5 外部 LSA。

图 9-82 显示了解决这个问题应遵循的流程图。

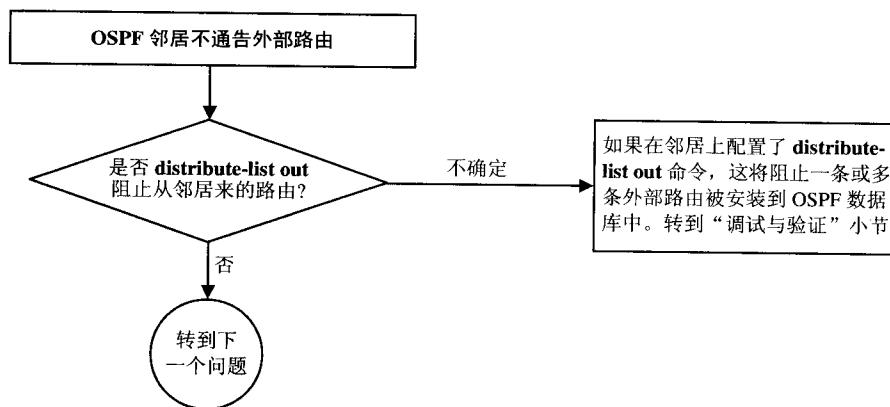


图 9-82 解决问题流程图

(1) 调试与验证

第一个逻辑步骤是检查配置。注意在前面的例子中关键字 `subnets` 阻止路由安装到路由表中。可是在这个例子中,分发表是阻止 131.108.3.0 重发布进 OSPF 的罪魁祸首。例 9-228

显示了 R1 的配置，它在 OSPF 中配置了 **distribute-list out** 语句，正是这个语句阻止了 132.108.3.0 进入 OSPF 数据库。

例 9-228 R1 的分发表阻止路由 132.108.3.0 安装到 OSPF 数据库中

```
R1#  
router ospf 1  
redistribute rip subnets  
network 131.108.1.0 0.0.0.255 area 2  
network 131.108.2.0 0.0.0.255 area 2  
distribute-list 1 out  
!  
access-list 1 permit 132.108.4.0 0.0.0.255
```

例 9-229 显示了 R1 开始为 132.108.4.0 但是不为 132.108.3.0 产生类型 5 外部 LSA，因为 131.108.3.0 网络在 access list1 中被明确地拒绝了。

例 9-229 确定由 R1 产生的外部 LSA

```
R1#show ip ospf database external 132.108.3.0  
  
OSPF Router with ID (131.108.1.2) (Process ID 1)  
  
R1#  
R1#show ip ospf database external 132.108.4.0  
  
OSPF Router with ID (131.108.1.2) (Process ID 1)  
  
Type-5 AS External Link States  
  
LS age: 1161  
Options: (No TOS-capability, DC)  
LS Type: AS External Link  
Link State ID: 132.108.4.0 (External Network Number )  
Advertising Router: 131.108.1.2  
LS Seq Number: 80000001  
Checksum: 0x550  
Length: 36  
Network Mask: /24  
Metric Type: 2 (Larger than any link state path)  
TOS: 0  
Metric: 1  
Forward Address: 0.0.0.0  
External Route Tag: 1  
  
R1#
```

(2) 解决方法

要解决这个问题，你必须要么去掉分发表使得 R1 为所有的 RIP 路由产生类型 5 外部 LSA 要么修改分发表使得它包含所有需要类型 5 LSA 的网络。当使用路由映射而不是分发表的时候，也会发生这个问题。在任何情况下，确信在访问列表中允许那些希望的网络。

例 9-230 显示了 R1 的配置，它在访问列表中允许网络 132.108.3.0，因而 R1 为这个网络产生类型 5 外部 LSA。

例 9-230 配置 R1 的分发表来允许网络 132.108.3.0

```
R1#
router ospf 1
 redistribute rip subnets
 network 131.108.1.0 0.0.0.255 area 2
 network 131.108.2.0 0.0.0.255 area 2
 distribute-list 1 out
!
access-list 1 permit 132.108.4.0 0.0.0.255
access-list 1 permit 132.108.3.0 0.0.0.255
```

在访问列表被修改之后，检查是否 R1 开始为 132.108.3.0 产生外部 LSA 了。例 9-231 显示在必要的配置改变之后 R1 开始为网络 132.108.3.0 产生外部 LSA 了。

例 9-231 确认 R1 现在为 132.108.3.0 产生外部 LSA 了

```
R1#show ip ospf database external 132.108.3.0

      OSPF Router with ID (131.108.1.2) (Process ID 1)

Type-5 AS External Link States

LS age: 1161
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 132.108.3.0 (External Network Number )
Advertising Router: 131.108.1.2
LS Seq Number: 80000001
Checksum: 0x550
Length: 36
Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 1

R1#
```

9.6 OSPF 中路由汇总疑难解析

本节讨论 OSPF 中的一项功能叫作路由汇总 (route summarization)。它的意思是如果存在邻近范围的地址，那么你可以不通告每一个网络，而是可以形成一组邻近的网络并且将这些网络汇总成一个、两个或少数几个块，并且通告这些块。这项功能有助于减小路由表的大小。减小路由表的大小也就减少了收敛的时间，同时提高了 OSPF 的性能。因此，汇总需要

手动地配置到路由器上。

OSPF 可以使用两种类型的汇总：

- 可以在 ABR 上进行的区域间汇总。
- 可以在 ASBR 上进行的外部汇总。

两个与 OSPF 中汇总相关的常见问题如下：

- 路由器不汇总区域间的路由。
- 路由器不汇总外部路由。

9.6.1 问题：路由器不汇总区域间路由——原因：没有在 ABR 上配置 area range 命令

你必须保证在相应的路由器上配置了 **area range** 命令。区域范围汇总仅能由 ABR 进行。在汇总中 ABR 不是为每个网络产生分离的 LSA，而是产生汇总 LSA 来覆盖这些地址范围。

有时，网络掩码配错了，汇总因为错误的配置而不工作。当配置 **area range** 命令的时候，确信汇总掩码是前缀掩码的形式而不是通配掩码的形式（也就是 255.255.255.0 而不是 0.0.0.255）。

图 9-83 显示了受到这个问题困扰的网络。在这个例子中，在 R1 上配置了命令 **area range**。这个命令仅仅只能配置在那些属于某个区域的路由器上，这个区域是产生汇总路由的来源地。另外，这个路由器必须是一个 ABR。

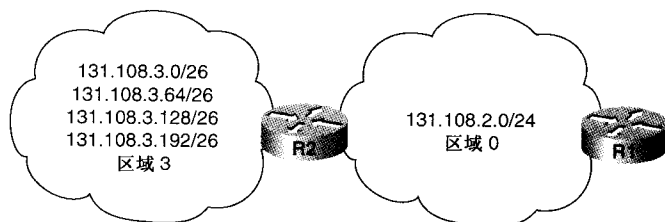


图 9-83 OSPF 网络，其中一个路由器不汇总区域间路由

图 9-84 显示了解决这个问题应遵循的流程图。

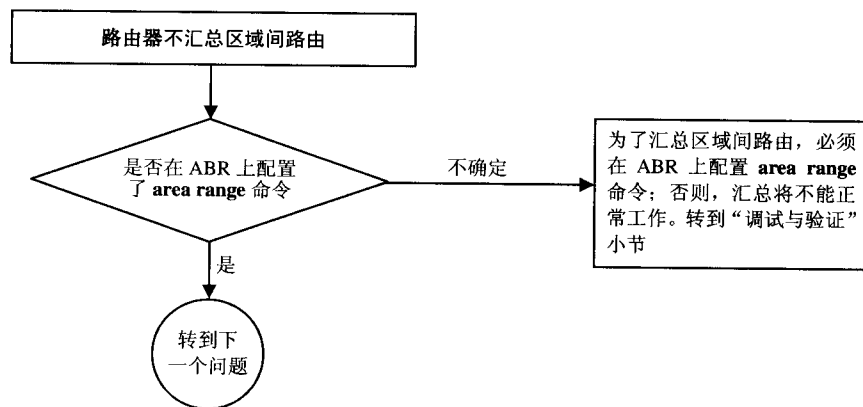


图 9-84 解决问题流程图

(1) 调试与验证

例 9-232 显示了 R1 为汇总区域 3 的路由而配置了 **area range** 命令。在例 9-232 中高亮显示的命令定义要进行汇总的范围。这表示在范围 131.108.3.0—255 中的任何地址将汇总到一条单一路由——131.108.3.0/24 中。掩码也是 255.255.255.0 的格式而不是 0.0.0.255。格式 0.0.0.255 用于访问列表范围中，但在 OSPF 中，这个格式应该是 255.255.255.0。

area range 命令的语法是正确的，但问题是 R1 不是一个 ABR。R1 确实包括在区域 0 中，但它不与区域 3 相连因而无法汇总区域 3 的路由。

例 9-232 R1 汇总区域 3 的路由

```
R1#
router ospf 1
network 131.108.2.0 0.0.0.255 area 0
area 3 range 131.108.3.0 255.255.255.0
```

有一个容易的方式来检查路由器是否正确地汇总路由。例 9-233 显示了 **show ip ospf** 的输出，表明区域 3 的范围是被动的 (passive)。被动表示没有这个区域的地址落在这个范围中，实际上，R1 确实拥有这个范围内的路由；可是因为 R1 不与区域 3 相连，所以这个范围表现为被动。也应注意在这个区域中接口数量为 0，表示这个路由器不与区域 3 相连。因此，汇总将不能正常工作。

例 9-233 区域 3 接口被定义为被动

```
R1#show ip ospf

Area 3
Number of interfaces in this area is 0
Area has no authentication
SPF algorithm executed 1 times
Area ranges are
131.108.3.0/24 Passive
```

因为汇总没有发生，所以 R1 接收了 4 条路由到它的路由表中而不是一条汇总路由。例 9-234 显示了 R1 接收 4 条通过 OSPF 区域间路由了解到的路由到路由表中。

例 9-234 R1 的路由表显示 4 条路由而不是一条汇总路由

```
R1#show ip route
...
O IA    131.108.3.0/26 [110/64] via 131.108.2.2, 00:01:35, Serial0
O IA    131.108.3.64/26 [110/64] via 131.108.2.2, 00:01:35, Serial0
O IA    131.108.3.128/26 [110/64] via 131.108.2.2, 00:01:35, Serial0
O IA    131.108.3.192/26 [110/64] via 131.108.2.2, 00:01:35, Serial0
```

(2) 解决方法

要解决这个问题，必须在 R2 上配置命令，R2 与区域 3 相连而且也是一个 ABR。例 9-235 显示了在 ABR 上配置 **area range** 命令，这个 ABR 也就是 R2。

例 9-235 在 R2 (ABR) 上配置 area range 命令

```
R2#
router ospf 1
 network 131.108.3.0 0.0.0.255 area 3
 network 131.108.2.0 0.0.0.255 area 0
 area 3 range 131.108.3.0 255.255.255.0
```

在配置了范围后，再一次检查范围状态来看它是主动的 (active) 还是被动的。如果它显示为主动的，路由器就会相应地开始汇总路由了。例 9-236 显示了 **show ip ospf** 的输出，表示区域 3 定义了区域范围并且处于主动状态。

例 9-236 确定区域 3 中的区域范围状态

```
R2#show ip ospf

Area 3
  Number of interfaces in this area is 4
  Area has no authentication
  SPF algorithm executed 1 times
  Area ranges are
    131.108.3.0/24 Active (64)
```

在确认 R2 确实汇总路由并且范围处于主动状态之后，检查 R1 路由表看是否 R1 仍然看到 4 条路由。例 9-237 显示了在汇总路由后，R1 开始接收一条路由而不是 4 条路由。

例 9-237 R1 接收一条单独的汇总路由

```
R1#show ip route
...
O IA    131.108.3.0/24 [110/64] via 131.108.2.2, 00:01:35, Serial0
```

9.6.2 问题：路由器不汇总外部路由——原因：没有在 ASBR 上配置 summary-address 命令

当任何外部的、静态的或者直连的协议被重发布进 OSPF 中的时候，一个 OSPF ASBR 产生外部 LSA。这些 LSA 在 ASBR 上产生。因此，当配置了汇总时，它总是应该在产生这些外部 LSA 的 ASBR 上配置；否则，汇总将不会正常工作。同样，汇总掩码语法和区域范围的一样——也就是 255.255.255.0 而不是 0.0.0.255（前缀掩码而不是通配掩码）。

图 9-85 显示了一个网络设置，其中一个路由器不能正确汇总外部路由。R2 是 ASBR，它重发布 RIP 路由到 OSPF 中。

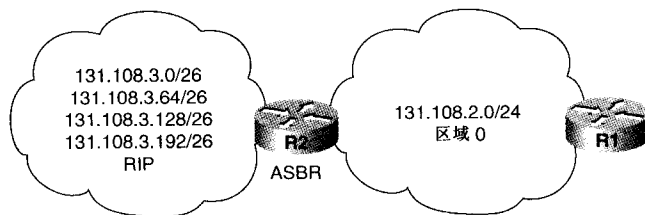


图 9-85 OSPF 网络受到路由器不能正确汇总外部路由问题的困扰

图 9-86 显示了解决这个问题应遵循的流程图。

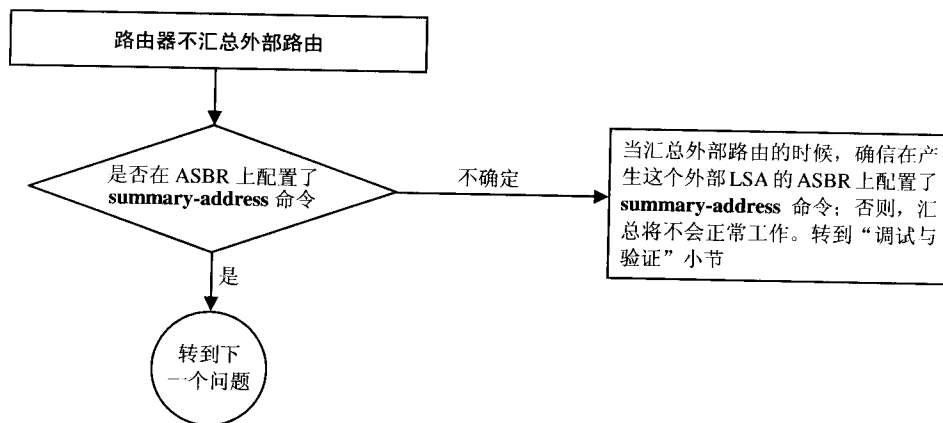


图 9-86 解决问题流程图

(1) 调试与验证

例 9-238 显示了在 R1 上的 **summary-address** 配置。注意 R1 不是 ASBR。也应该注意到范围使用格式 255.255.255.0 而不是 0.0.0.255，如同在前面问题中解释的一样。另外，在前面的例子中，命令 **area range** 被用于汇总区域路由，但那条命令不能在这使用，因为这些都是外部路由。为了汇总这些外部路由，必须使用 **summary-address** 命令。

例 9-238 R1 尽管不是 ASBR，它仍被配置来进行地址汇总

```

R1#
router ospf 1
network 131.108.2.0 0.0.0.255 area 0
summary-address 132.108.3.0 255.255.255.0
  
```

例 9-239 显示了 **show ip ospf summary-address** 的输出，表明这个汇总路由的度量值是 16777215——这是无限大，因为外部 LSA 度量值是 24 比特长，而 2^{24} 等于 16 777 215。一个无限大的度量值表示没有可用的地址属于这个范围。

例 9-239 汇总路由有一个无限大的度量值

```

R1#show ip ospf summary-address

OSPF Process 1, Summary-address

132.108.3.0/255.255.255.0 Metric 16777215, Type 0, Tag 0
R1#
  
```

(2) 解决方法

要解决这个问题，可在 R2 上配置汇总，R2 是一个 ASBR。例 9-240 显示了现在在 OSPF ASBR 上配置了汇总，这个 ASBR 就是 R2。

例 9-240 在相应的路由器，也就是 ASBR 上配置地址汇总

```
R2#
router ospf 1
network 131.108.2.0 0.0.0.255 area 0
summary-address 132.108.3.0 255.255.255.0
!
router rip
network 132.108.0.0
```

确信从 R1 上去掉了 `summary-address` 命令。在 R2 上配置了 `summary-address` 命令之后，`show ip ospf summary-address` 的输出将被再次检查看是否有正确的度量值。例 9-241 给出了 `show ip ospf summary-address` 的输出，表明范围可用并且有一个正确的度量值。这个汇总路由的度量值是在汇总范围内所有地址的最大度量值。这与 RFC 2178 一致。在 RFC 1583 中，汇总路由的度量值通常是在汇总范围内所有地址的最小度量值。

例 9-241 确认汇总地址范围现在可用了

```
R2#show ip ospf summary-address
OSPF Process 1, Summary-address

132.108.3.0/255.255.255.0 Metric 5, Type 0, Tag 0
R2#
```

9.7 CPUHOG 问题疑难解析

当 OSPF 形成邻接体关系时，它向它的邻居泛洪所有的链路状态更新分组。有时，泛洪过程要花费不少时间，这依据路由器的资源而不同。当泛洪使用了路由器大部分资源的时候，路由器的 CPU 变得过于忙碌，这时在日志中就会出现 CPUHOG 消息。

CPUHOG 消息通常出现在两个重要的阶段：

- 邻居形成过程。
- LSA 刷新过程。

本节讨论下面这两个 OSPF 实例的可能的解决方法：

- CPUHOG 消息出现于邻接体关系形成过程中。
- CPUHOG 消息出现在 LSA 刷新阶段中。

9.7.1 问题：CPUHOG 消息出现在邻接体关系形成过程中——原因：路由器没有运行分组调速代码

当 OSPF 形成邻接体关系的时候，它向邻居泛洪它所有的链路状态分组。这个泛洪过程有时会占用大量 CPU 资源。在 12.0T 以前的 Cisco IOS 软件版本中也不支持分组调速（packet-pacing）功能，分组调速表示一个路由器将试图在一条链路上尽它可能快速地发送数

据。如果一条链路是慢速链路或者在链路另一边的路由器不能足够快地回应，这就导致了 LSA 的重传并且最终导致出现 CPUHOG 消息。分组调速在 LSA 更新中间增加了一个步速间隔。它不是一次泛洪所有的分组，而是以一个几毫秒的时间间隔来发送分组。图 9-87 显示了解决这个问题应遵循的流程图。

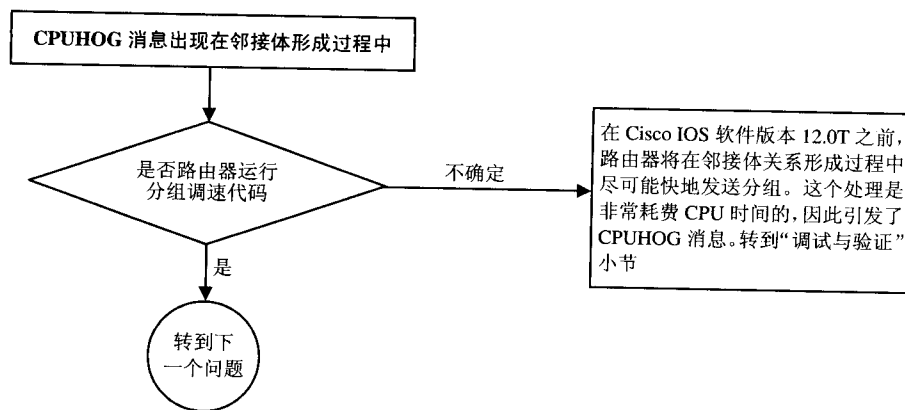


图 9-87 解决问题流程图

(1) 调试与验证

CPUHOG 消息可以在邻接体形成过程中在一个路由器的控制台上看到或者之后用 `show log` 命令来检查。例 9-242 显示了一个路由器上的日志消息显示了 CPUHOG。

例 9-242 日志消息显示 OSPF 路由器的 CPUHOG 消息

```

R1#show log
%SYS-3-CPUHOG: Task ran for 2424 msec (15/15), process = OSPF Router
%SYS-3-CPUHOG: Task ran for 2340 msec (10/9), process = OSPF Router
%SYS-3-CPUHOG: Task ran for 2264 msec (0/0), process = OSPF Router
  
```

(2) 解决方法

分组调速在分组发送间引入了一个 33 毫秒的延迟，在重传过程内引入了 66 毫秒的延迟。这个步速间隔减少了 CPUHOG 消息，并且邻接体关系形成也更快了。在 Cisco IOS 软件版本 12.0T 和更高版本中这个特性是缺省具备的。

这个特性在 Cisco IOS 软件 12.0T 以前的版本中不提供，如果你运行的 Cisco IOS 软件代码版本早于 12.0T 并且在邻接体形成过程中看到了 CPUHOG 消息，那么请升级到至少 Cisco IOS 软件版本 12.0T 或更高版本代码，通过使用分组调速来解决这个问题。

9.7.2 问题：CPUHOG 消息出现在 LSA 刷新阶段——原因：路由器没有运行 LSA 组调速代码

当 Cisco IOS 软件代码不是版本 12.0T 或者更高版本的时候发生这个问题。在 Cisco IOS 软件版本 12.0 中，LSA 组调速 (LSA group pacing) 特性被引入来消除每 30 分钟出现的这个 CPU 问题。

在前面版本的 Cisco IOS 软件中,所有的 LSA 每 30 分钟刷新一次来同步所有 LSA 的年龄。因此,每 30 分钟会有一次重要的泛洪来在同一时刻刷新所有的 LSA。这次泛洪导致了每 30 分钟出现一次的 CPUHOG 消息。想像一下,数千条 LSA 在同一时刻被刷新是怎样一种情况。

图 9-88 显示了解决这个问题的流程图。

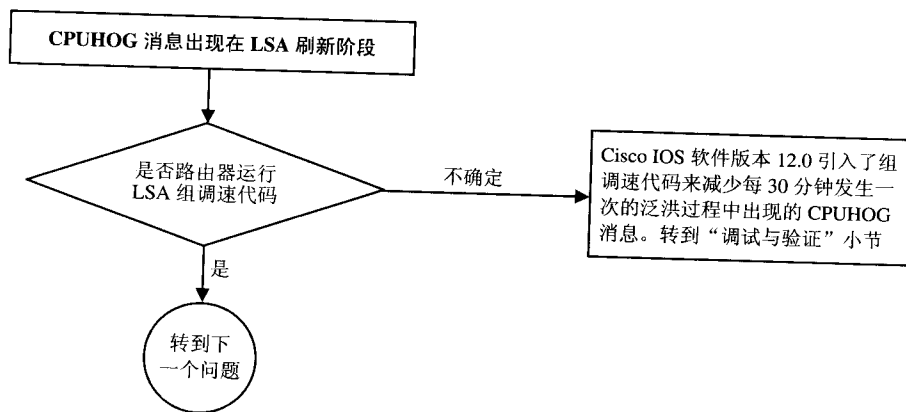


图 9-88 解决问题流程图

(1) 调试与验证

例 9-243 显示了在路由器日志中每 30 分钟出现的 CPUHOG 问题。

例 9-243 路由器每 30 分钟看到一次 CPUHOG 消息

```

R1#show log
%SYS-3-CPUHOG: Task ran for 2424 msec (15/15), process = OSPF Router
%SYS-3-CPUHOG: Task ran for 2340 msec (10/9), process = OSPF Router
%SYS-3-CPUHOG: Task ran for 2264 msec (0/0), process = OSPF Router
  
```

(2) 解决方法

LSA 组调速功能每个周期间隔(缺省值为 4 分钟)检查一下 LSA, 并且仅仅刷新那些过了它们刷新时间的 LSA。这是通过将一个大量 LSA 泛洪分段成多个小的 LSA 泛洪而减少每一次泛洪数量的有效手段, 对这一功能不需要额外的配置, 但是对大量的 LSA (通常为 10,000 或者更多), 则建议使用小的间隔(例如 2 分钟); 对于几百条 LSA, 则可以使用一个大的间隔, 如 20 分钟。

如果有 10,000 条 LSA 需要更新, 那么保持刷新间隔小一些, 每 2 分钟或者每 4 分钟检查一次 LSA 看有多少 LSA 已经达到了刷新间隔, 这个间隔是 30 分钟。使用这种频率检查的好处是每 2 分钟或者每 4 分钟只有少量 LSA 需要被刷新, 这不会引起大的 LSA 更新风暴。如果 LSA 的数量很小, 那么每 2 分钟刷新一次或者每 20 分钟刷新一次都不重要了。这就是为什么最好是增加计时器使得所有能被立刻更新的 LSA 数量上很少的原因。

例 9-244 配置 LSA 刷新间隔

```

R1(config)#router ospf 1
R1(config-router)#timer lsa-group-pacing ?
<10-1800> Interval between group of LSA being refreshed or maxaged
  
```

9.8 OSPF 中按需拨号路由 (DDR)

问题疑难解析

本节讨论与 DDR 相关的问题。当 OSPF 配置在一条 DDR 链路上时，确信禁止 OSPF Hello 分组，因为 OSPF 在点到点链路上每 10 秒发送一次 Hello 分组。

与在 DDR 链路上 OSPF 相关的最常见问题如下：

- 问题：OSPF Hello 分组启动链路。
- 问题：OSPF Hello 分组不能通过这条链路*。
- 问题：请求线路持续启动链路。

注意：*OSPF Hello 分组不能通过链路的问题在第 9.2 节“OSPF 邻居关系疑难解析”中已经解决，参考其中关于这个问题的解决方法。

9.8.1 问题：OSPF Hello 分组启动链路——原因：OSPF Hello 分组被允许作为关注流量

当为了拨号备份的目的而在 DDR 链路上运行 OSPF 的时候，定义一个访问列表来明确地定义关注的流量 (interesting traffic)。OSPF 使用组播地址 224.0.0.5 来发送 Hello 分组。这个地址必须在访问列表中被拒绝，这样 OSPF 就不会每 10 秒钟启动一次链路了。

图 9-89 显示了一个正面临这个问题的网络。

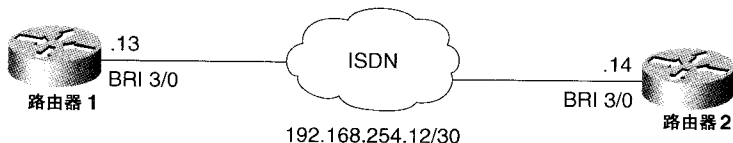


图 9-89 OSPF 网络正面临一个没完没了的 DDR 链路问题

图 9-90 显示了解决这个问题应遵循的流程图。

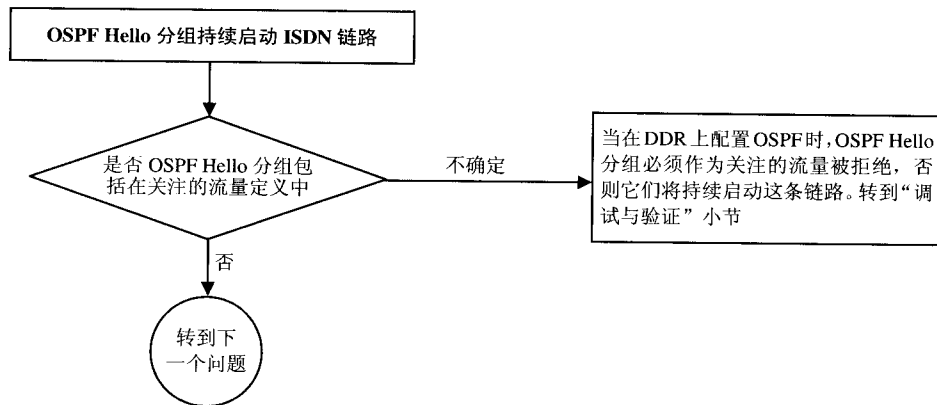


图 9-90 解决问题流程图

(1) 调试与验证

例 9-245 显示了 R1 上的可能引起这个问题的配置。在这个配置中，仅仅是 TCP 流量被拒绝。换句话说，TCP 流量不会启动这条链路，但是任何其他 IP 流量可以这样做。

例 9-245 R1 的访问列表仅仅拒绝 TCP 流量

```
R1#
interface BRI3/0
ip address 192.168.254.13 255.255.255.252
encapsulation ppp
dialer map ip 192.168.254.14 name R2 broadcast 57654
dialer-group 1
isdn switch-type basic-net3
ppp authentication chap

access-list 100 deny tcp any any
access-list 100 permit ip any any
dialer-list 1 protocol ip list 100
```

例 9-246 中 **show dialer** 命令的输出表示链路启动的原因是 OSPF 组播 Hello 分组。

例 9-246 确定为什么 DDR 链路被持续启动

```
R1#show dialer
BRI1/1:1 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (2 secs)
Dialer state is data link layer up
Dial reason: ip (s=192.168.254.13, d=224.0.0.5)
Current call connected 00:00:08
Connected to 57654 (R2)
```

(2) 解决方法

要解决这个问题，需要拒绝关注流量中的 OSPF Hello 分组。例 9-247 显示了在 R1 上正确的配置变化。在这个配置中，所有目的地址为 224.0.0.5 的流量都被拒绝。这表示在这条点到点链路上的 OSPF Hello 分组或者更新分组在这个配置改变后都不能启动链路了。

例 9-247 配置 R1 的访问列表来拒绝 OSPF 组播 Hello 分组

```
R1#
access-list 100 deny tcp any any
access-list 100 deny ip any 224.0.0.5
access-list 100 permit ip any any
dialer-list 1 protocol ip list 100
```

不需要将 224.0.0.6 放在访问列表中，因为仅仅 DR 使用这个地址。也因为在这条链路上没有 DR，所以不需要在访问列表中拒绝这个地址。

9.8.2 问题：请求线路持续启动链路

OSPF 请求线路功能在 Cisco IOS 软件版本 11.2 中引入。这个功能在一条链路上形成 OSPF

邻接体关系，接着在此之后使得第 2 层关闭以使得在这条链路上保持 OSPF 邻接体关系的同时减少开销。如果这条链路被持续启动，它就使得请求线路的目的失败了。

这个问题最常见的可能原因如下：

- 在网络中存在链路抖动。
- 网络类型被定义为广播类型。
- PPP 主机路由被重发布进 OSPF 数据库中。
- 其中某个路由器不具备请求线路功能。

1. 请求线路持续启动链路——原因：在网络中存在链路抖动

请求线路来启动一条链路最常见的原因是链路抖动的存在。当一条链路在网络中的任何一部分一会连通一会断开的时候，就会发生链路抖动。这引起数据库中信息的改变，并且 OSPF 必须启动这条链路同时与在这条请求线路上的邻居一起来刷新它们的数据库。这在图 9-91 的网络设置中显示。一条链路在区域 0 中抖动，并引起区域 0 的 SPF 计算过程。因为 R1 也是区域 0 的一部分，R1 将运行 SPF 算法并且接着启动这条通往 R2 的请求线路来通知 R2 这个变化。

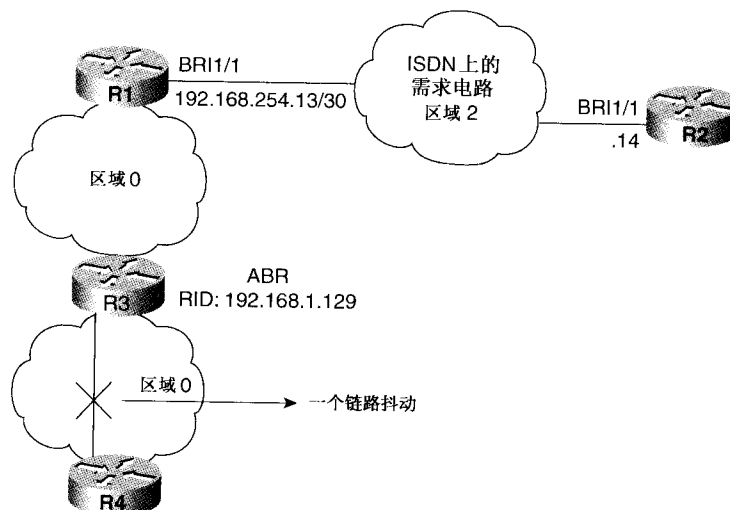


图 9-91 OSPF 网络被一个由请求线路引起的惯性的启动链路问题所困扰

图 9-92 显示了解决这个问题应遵循的流程图。

(1) 调试与验证

例 9-248 显示了 **show dialer** 的输出，表明最后一次呼叫连接是由 OSPF Hello 分组引起的并且表示这个呼叫已经持续连续 8 秒钟了。

例 9-248 确定在拨号接口上的呼叫历史

```
R1#show dialer
BRI1/1:1 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (2 secs)
Dialer state is data link layer up
Dial reason: ip (s=192.168.254.13, d=224.0.0.5)
Current call connected 00:00:08
Connected to 57654 (R2)
```

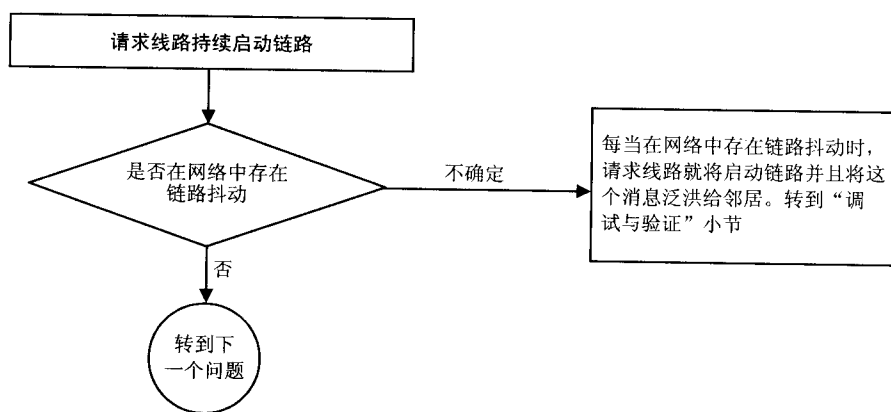


图 9-92 解决问题流程图

当 OSPF 中的一条链路抖动时，很容易用 **debug ip ospf monitor** 命令发现。这个命令的输出显示了区域中抖动的确切的 LSA。例 9-249 显示了 **debug ip ospf monitor** 的输出，表明因为在区域 0 中存在链路抖动，所以有一个路由器 LSA 被一个路由器 ID 为 192.168.1.129 的路由器重新产生，结果导致了在 OSPF 数据库中的变化。

例 9-249 debug 揭示了链路抖动现象发生的罪魁祸首

```

R1# debug ip ospf monitor
OSPF: Schedule SPF in area 0.0.0.0
      Change in LS ID 192.168.1.129, LSA type R,
OSPF: schedule SPF: spf_time 1620348064ms wait_interval 10s
  
```

(2) 解决方法

对请求线路来说，每当 OSPF 数据库中发生一个变化，它就必须启动链路并且将这个变化泛洪给邻居，这种现象是很正常的。在某些区域中的链路抖动将重新产生一个路由器 LSA，它将引起为这个网络重新产生汇总 LSA。

因为链路抖动几乎是永远不可避免的，因此在这种情况下，一种解决方法是将发生链路抖动的区域尽可能地隔离开。换句话说，就是尝试将它作为一个端区域或者完全端区域来运行。

当区域被定义为端区域的时候，就没有外部链路抖动可以启动这条拨号链路了。当区域配置为完全端区域的时候，就没有区域间抖动可以启动这条拨号链路了。这是因为当区域被定义为端区域之后，就没有外部 LSA 可以进入这个端区域；也就没有外部抖动可以在端区域中感受到了。类似地，当一个区域是完全端区域的时候，就没有汇总 LSA 可以在这个区域中存在于；汇总 LSA 中间的任何变化都不会在完全端区域中被感受到。

例 9-250 显示了区域 2 被配置成了完全端区域，因此没有区域间链路抖动可以启动这条链路。

例 9-250 配置区域 2 成为完全端区域网络来阻止区域间链路抖动启动这条链路

```

On R1 and R2:
router ospf 1
 network 192.168.254.0 0.0.0.255 area 2
 area 1 stub no-summary
  
```

R2 上不真正需要命令 **no-summary**。在 R1 上配置这条命令就足够了，但是如果在两边都配置了这条命令的话也不会产生任何危害。

2. 请求线路持续启动链路——原因：网络类型被定义为广播类型

当网络类型被定义为广播类型的时候，OSPF Hello 分组没有被禁止。可是，没有泛洪每隔 30 分钟发生一次，因为配置了请求线路。因此，通过配置网络类型为广播类型，获得的仅仅是最佳的泛洪，但 Hello 分组仍然启动链路。

图 9-93 显示了解决这个问题应遵循的流程图。

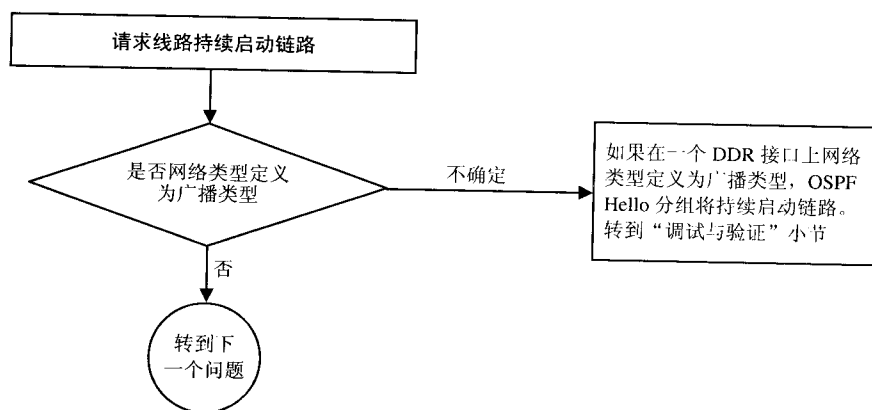


图 9-93 解决问题流程图

(1) 调试与验证

PPP 链路的缺省网络类型是点到点类型，但是有人手工地将网络类型改成了广播类型。

例 9-251 显示了 R1 上的配置，其中显示网络类型定义为广播类型。

例 9-251 R1 的 BRI1/1 接口被定义为广播类型

```

R1#
interface BRI1/1
 ip address 192.168.254.13 255.255.255.0
 ip ospf network broadcast
!
  
```

例 9-252 显示了在 BRI1/1 接口上运行 **show ip ospf interface** 的输出，表示 Hello 分组没有被禁止。

例 9-252 show ip ospf interface 命令输出揭示了 OSPF Hello 分组没有被禁止

```

R1#show ip ospf interface bri1/1
BRI1/1 is up, line protocol is up
Internet Address 192.168.254.13/24, Area 1
Process ID 1, Router ID 192.168.254.13, Network Type BROADCAST, Cost: 64
Transmit Delay is 1 sec, State BDR, Priority 240
Designated Router (ID) 192.168.254.14, Interface address 192.168.254.14
Backup Designated router (ID) 192.168.254.13, Interface address 192.168.254.13
Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
Hello due in 00:00:21
  
```

这明显表示 OSPF Hello 分组将每隔一个 Hello 间隔在链路上发送一次。例 9-253 显示了拨号备份链路被 OSPF Hello 分组启动了。

例 9-253 OSPF Hello 分组启动拨号备份链路

```
R1#show dialer
BRI1/1:1 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (2 secs)
Dialer state is data link layer up
Dial reason: ip (s=192.168.254.13, d=224.0.0.5)
Interface bound to profile Di1
Current call connected 00:00:08
Connected to 57654 (R2)
```

(2) 解决方法

要解决这个问题,可改变网络类型回到点到点类型。这将使 OSPF Hello 分组不再启动链路。

例 9-254 显示了解决这个问题的新配置。网络类型没有在配置中显示出来,因为缺省时,在 BRI 链路上的网络类型为点到点类型。

例 9-254 缺省情况下, R1 和 R2 之间的 BRI 现在配置为点到点类型

```
R1#
interface BRI1/1
ip address 192.168.254.13 255.255.255.0
no ip ospf network broadcast
!

R2#
interface BRI0/1
ip address 192.168.254.14 255.255.255.0
no ip ospf network broadcast
!
```

例 9-255 显示了在改变网络类型之后,在 BRI1/1 接口上运行 **show ip ospf interface** 的输出,表示 BRI1/1 接口在请求线路上禁止向邻居发送 Hello 分组。有三个重要的事情在这个例子中被高亮显示:

- 这条链路现在作为点到点网络运行。
- 这条链路被配置为请求线路。
- 在这条链路上禁止 OSPF Hello 分组。

例 9-255 在 R1 的 BRI1/1 接口上禁止 OSPF Hello 分组

```
R1#show ip ospf interface BRI1/1
BRI1/1 is up, line protocol is up
Internet Address 192.168.254.13/24, Area 1
Process ID 1, Router ID 192.168.254.13, Network Type POINT_TO_POINT, Cost: 64
```

(待续)

```

Configured as demand circuit.
Run as demand circuit.
DoNotAge LSA allowed.
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 00:00:06
Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 192.168.254.14 (Hello suppressed)
Suppress hello for 1 neighbor(s)

```

3. 请求线路持续启动链路——原因: PPP 主机路由被重发布进 OSPF 数据库中

当在链路上使用 PPP 封装的时候,它为远端邻居的 IP 地址安装一条主机路由。这对 PPP 封装来说很正常。在 OSPF 中,这条主机路由不会重发布进数据库中。特别地,当 OSPF 在一条链路上作为请求线路运行的时候,在数据库中包含这条主机路由可能引起不断启动链路这样的问题。图 9-94 显示了一个网络,其中长期存在请求线路启动另一条链路的情况,这是 PPP 主机路由被重分布进 OSPF 数据库的结果。

R1 运行 RIP 和 OSPF 协议并且将 RIP 重发布进 OSPF。主机路由重发布进 OSPF 数据库,因为 RIP 拥有这条主机路由,该路由是通过 PPP 安装的。RIP 将这条路由作为一条外部路由引入,这就引起了链路抖动。

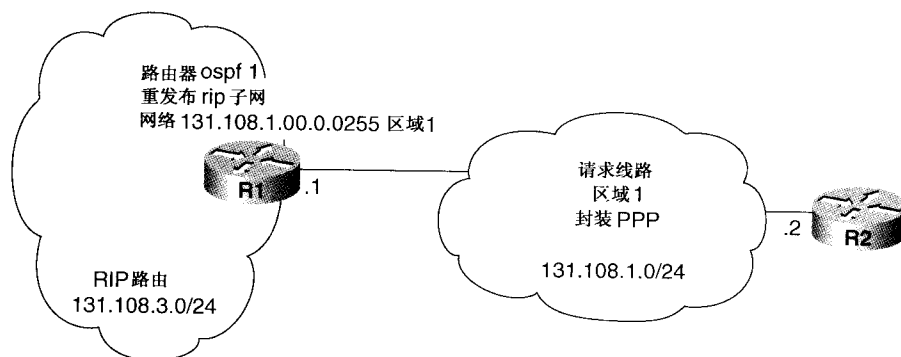


图 9-94 网络被请求线路引起的习惯性的链路启动问题所困扰,这是由于 PPP 主机路由重发布引发的

图 9-95 显示了解决这个问题应遵循的流程图。

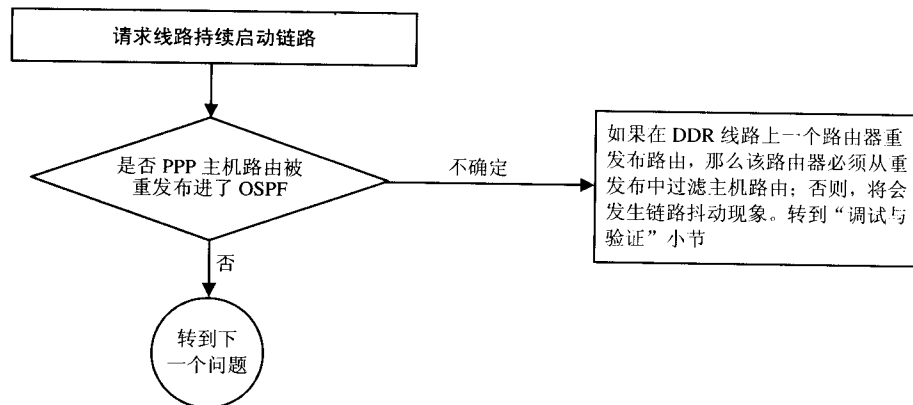


图 9-95 解决问题流程图

(1) 调试与验证

首先,通过检查 R1 的配置确认确实是 PPP 封装。例 9-256 显示了 R1 上的配置。这个配置揭示了 R1 的 BRI1/1 运行 PPP 封装。R1 也重发布 RIP 路由到 OSPF 中;RIP 有一个 **network 131.108.0.0** 语句,因此任何在范围 131.108.0.0 中的直连路由也将被 RIP 拥有。这也包括了 PPP 主机路由。

例 9-256 R1 重发布 RIP 路由到 OSPF 中

```
R1#
interface BRI1/1
encapsulation PPP
ip address 131.108.1.1 255.255.255.0
router ospf 1
 redistribute rip subnets
 network 131.108.1.0 0.0.0.255 area 1
!
router rip
 network 131.108.0.0
```

例 9-257 显示了为 131.108.1.2 安装了一条掩码为/32 的路由到 R1 的路由表中。

例 9-257 R1 为 R2 安装一条 PPP 主机路由

```
R1#show ip route 131.108.1.2
Routing entry for 131.108.1.2/32
  Known via "connected", distance 0, metric 0 (connected, via interface)
  Routing Descriptor Blocks:
    * directly connected, via BRI1/1
      Route metric is 0, traffic share count is 1
```

因为 RIP 对范围 131.108.0.0 内的所有直连路由拥有所有权,所以它也拥有这条直连主机路由。当 RIP 被重发布进 OSPF 中的时候,这条主机路由作为一条外部路由重发布进 OSPF 中。例 9-258 显示了这条直连路由被重发布进 OSPF 数据库中,因为 RIP 有一条 **network** 语句包括这条主机路由。

例 9-258 主机路由作为外部路由重发布进 OSPF 数据库

```
R1#show ip ospf database external 131.108.1.2

OSPF Router with ID (131.108.3.1) (Process ID 1)

Type-5 AS External Link States

LS age: 298
Options: (No TOS-capability, DC)
LS Type: AS External Link
Link State ID: 131.108.1.2 (External Network Number )
Advertising Router: 131.108.3.1
LS Seq Number: 80000001
```

(待续)

```
Checksum: 0xDC2B
Length: 36
Network Mask: /32
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 20
Forward Address: 0.0.0.0
External Route Tag: 0
```

(2) 解决方法

由于 RIP 有一个 131.108.0.0 的 **network** 语句而引发了这个问题。根据定义, 任何落在这个地址范围内的接口都被 RIP 通告。当 PPP 连接建立起来的时候, 一条掩码为/32 的主机路由被 PPP 引入进来。因为 BRI 接口地址是 131.108.1.0/24, 所以这条主机路由落入到了 131.108.0.0 的范围内。因为 RIP 路由被重发布进 OSPF, 这个掩码/32 也被重发布进 OSPF。当链路断开的时候, 这个/32 消失了, 并且 OSPF 也从数据库中把它去掉了, 这就导致了一个会聚事件。

当 OSPF 去掉这条路由的时候, 在 OSPF 数据库中发生了一个变化, 使得再次启动请求线路接口。

这个问题可以通过三种方式解决:

- 如果运行 Cisco IOS 软件版本 11.3 或更高版本的话, 在 **interface** 命令下使用 **no peer neighbor-route** 语句。
- 在拨号链路上分配一个不同的主网络号。
- 在重发布过程中过滤主机路由。

第一个解决方法依赖于 Cisco IOS 软件版本 11.3 或者更高版本。例 9-259 显示了在 R1 上的解决这个问题的新配置。

例 9-259 从 R1 上去掉 PPP 主机路由

```
R1# interface BRI1/1
ip address 131.108.1.1 255.255.255.0
encapsulation ppp
no peer neighbor-route
```

在使用这个命令后, R1 将不在它的路由表中安装路由 131.108.1.2/32; 因此这个问题将不再发生。

第二种解决方法难于实现, 但是如果它可以完成的话, RIP 将不再通告被 PPP 安装的主机路由。

第三种解决方法是最希望的解决方法, 因为它不依赖于任何特定的 Cisco IOS 软件版本。例 9-260 显示了使用这种方法解决这个问题的新配置。在这个配置中, RIP 被重发布进 OSPF, 同时通过一个路由映射过滤直连主机路由。路由映射通常在路由重发布过程中过滤路由。路由映射调用一个访问列表; 在这个例子中, access list 1 有被过滤的直连主机路由。

例 9-260 在重发布中过滤主机路由

```
R1#
router ospf 1
```

(待续)

```

redistribute rip subnets route-map rip_filter
network 131.108.1.0 0.0.0.255 area 1
!
router rip
network 131.108.0.0
!
route-map rip_filter permit 10
match ip address 1
!
access-list 1 permit 131.108.3.0 0.0.0.255

```

4. 请求线路持续启动链路——原因：其中某个路由器不具备请求线路功能

如果网络中的任何一个路由器不能够理解 DNA LSA 的话，那么请求线路将启动链路。DNA LSA 在第 8 章中详细讨论。通常，这一现象发生在两种情况下：

- 路由器是一个非 Cisco 路由器，它没有请求线路的功能。
- 路由器是一个 Cisco 路由器，但它运行的 Cisco IOS 软件版本早于 11.2 因而不支持请求线路。

图 9-96 显示了一个网络设置，其中一条请求线路习惯性地启动另一条链路，因为其中的一个路由器不具备请求线路功能。R1 运行 Cisco IOS 软件版本 11.1.20 因而没有请求线路功能。所有其他路由器都在区域 1 中。

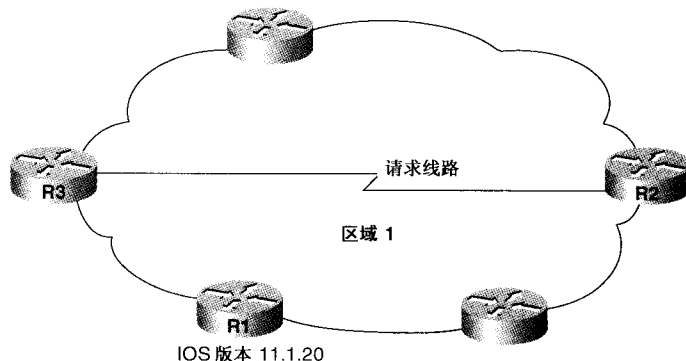


图 9-96 网络中有一个路由器不支持请求线路

图 9-97 显示了解决这个问题应遵循的流程图。

(1) 调试与验证

例 9-261 显示了 R3 的 BRI 接口定义为请求线路，但是在区域 1 中不允许 DoNotAge LSA。

例 9-261 在区域 1 中不允许 DNA LSA，但是 Hello 分组仍然被禁止

```

R3#show ip ospf interface BRI1/1
BRI1/1 is up, line protocol is up
Internet Address 131.108.1.1/24, Area 1
Process ID 1, Router ID 131.108.3.3, Network Type POINT_TO_POINT, Cost: 64
Configured as demand circuit.
Run as demand circuit.
DoNotAge LSA not allowed (Number of DCbitless LSA is 1).

```

(待续)

```

Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:01
Neighbor Count is 1, Adjacent neighbor count is 1
Adjacent with neighbor 131.108.2.2 (Hello suppressed)
Suppress hello for 1 neighbor(s)

```

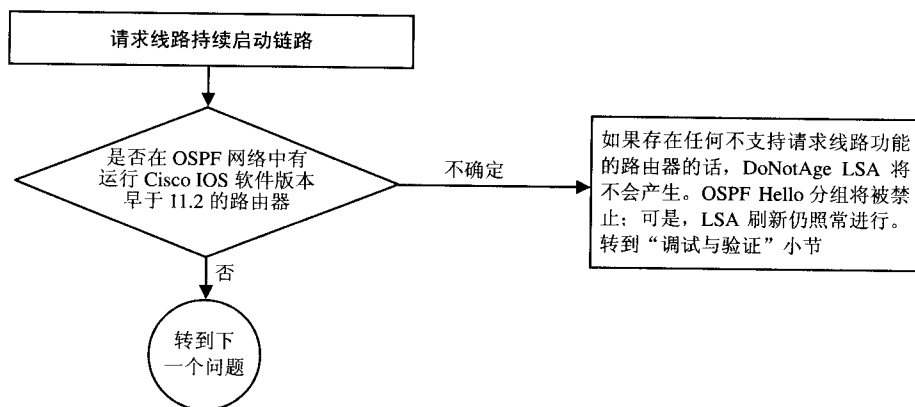


图 9-97 解决问题流程图

因为 OSPF Hello 分组被禁止, 所以在 Hello 间隔中链路将不会被启动; 可是, 当到了要刷新 LSA 的时候, 链路将在 LSA 泛洪过程中被启动。例 9-262 显示了 ISDN 链路被启动了, 因为一个 LSA 到达了刷新时间并且这条信息被泛洪进了请求线路链路中。输出显示 OSPF Hello 分组启动了链路。

例 9-262 LSA 泛洪启动了 ISDN 链路

```

R3#show dialer
BRI1/1:1 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (2 secs)
Dialer state is data link layer up
Dial reason: ip (s=131.108.1.1, d=224.0.0.5)
Interface bound to profile Di1
Current call connected 00:00:08
Connected to 57654 (R2)

```

(2) 解决方法

因为 R1 运行一个老的代码, 它不支持请求线路, 所以发生这个问题。这种情况在 RFC 1793 中关于请求线路的部分中已提及。解决方法是不产生任何 DoNotAge LSA, 因为有请求线路能力的路由器将不会正确解释它们。

对存在的这个情况有多种解决方法。最简单的就是升级 R1, 使它具有拥有请求线路能力的代码。

这个例子讲述了在一个单独区域中请求线路启动链路的情况。当存在多个区域的时候, 会发生一个相似的问题。图 9-98 显示了另一个产生这个问题的设置。R3 将会启动 ISDN 链路, 因为 R1 不能够理解 DNA LSA。这个特殊情况在第 8 章中“请求线路”小节中有详细

的讨论。

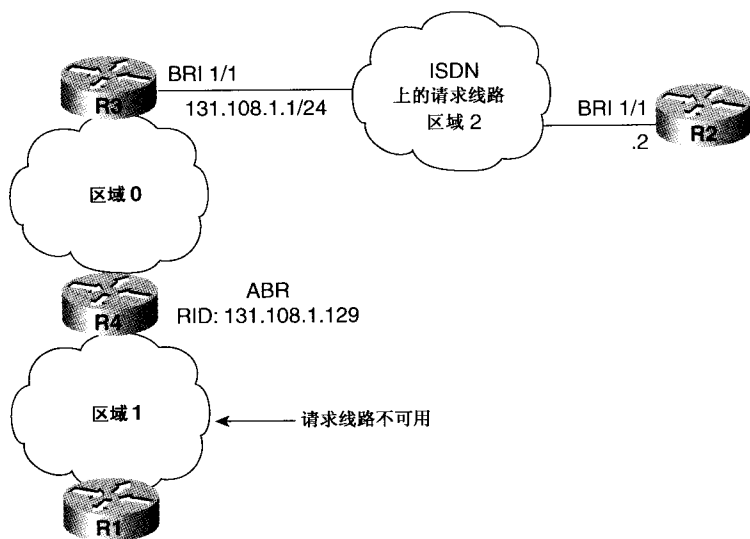


图 9-98 在多个区域中请求线路启动链路

在这个情况中的解决方法是配置区域 2 为一个完全端区域。通过定义区域 2 为完全端区域，就没有外部或者汇总 LSA 可以进入到区域 2 中。这表示指示 (indication) LSA 也将被阻止进入区域 2 中。第 8 章详细讨论了这个指示 LSA。例 9-263 显示了改变区域 2 为一个完全端区域的配置。区域 2 中的所有路由器都必须配置为端区域；否则，没有路由器会与 R1 形成邻接体关系。

例 9-263 配置区域 2 为端区域

```
R3#
router ospf 1
network 131.108.1.0 0.0.0.255 area 2
area 2 stub no-summary
```

命令 **no-summary** 仅仅需要增加到 R3 上，但如果增加它到区域 2 中的其它路由器上也不会造成任何伤害。区域 2 中的所有其他路由器必须至少配置有 **area 2 stub** 命令。

9.9 SPF 计算和路由抖动疑难解析

本节解释隐藏在 OSPF 中的路由抖动和 SPF 计算背后的常见原因。每当网络拓扑中有一个改变，OSPF 就运行 SPF 算法来再一次计算最短路径优先树。存在于 OSPF 网络云中的不稳定的链路将引起 SPF 计算经常发生。

本节讨论因为如下原因在网络中时常运行 SPF 算法的问题：

- 在网络中存在接口抖动。
- 在网络中存在邻居抖动。
- 路由器 ID 重复了。

9.9.1 SPF 经常运行——原因：在网络中存在接口抖动

这是 OSPF 中的一个常见问题。每当在区域中存在链路抖动的时候，OSPF 就运行 SPF 算法。因此，如果一个网络有不稳定的链路，它可能引起经常性的 SPF 运行问题。SPF 自身不是问题，因为 OSPF 仅仅只是通过计算 SPF 来调整数据库中的变化。如果网络中只有少数几个路由器，就会发生真正的问题，而且，一个经常性的 SPF 运行问题可能引起路由器中的 CPU 失效。链路抖动问题显示在图 9-99 中。因为 R1 也包含在区域 0 中，所以在区域 0 中的任何链路抖动将会引发区域 0 中的所有路由器运行 SPF 算法。

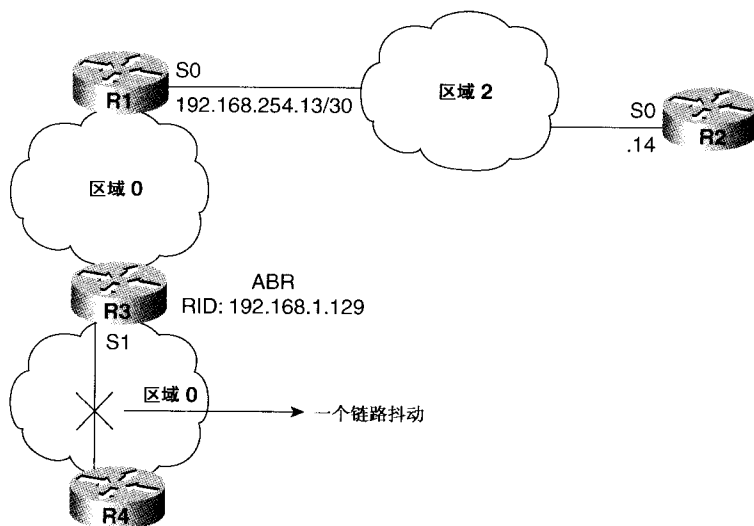


图 9-99 在区域 0 中的链路抖动引起 SPF 计算问题

图 9-100 显示了解决这个问题应遵循的流程图。

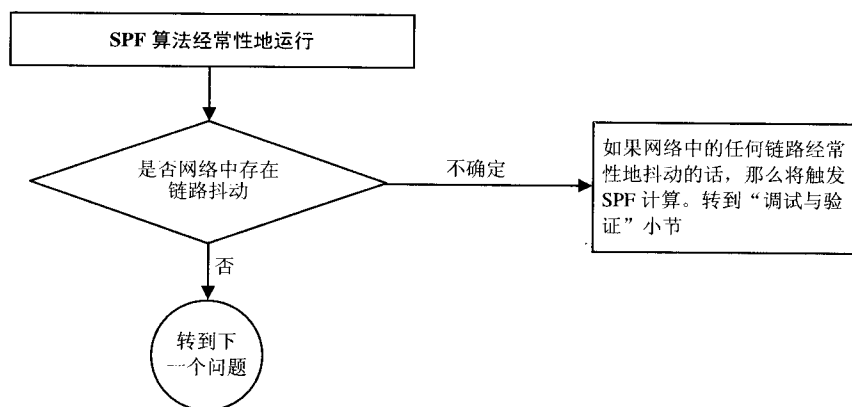


图 9-100 解决问题流程图

(1) 调试与验证

在区域中的链路抖动引起 SPF 运行。如果链路经常抖动的话，这可能增加在区域中 SPF 计算的次数。一个恒定数目的 SPF 计算不是问题，但是如果这个数目不断增加，那么这就是

一个问题的征兆。

例 9-264 显示了 **show ip ospf** 的输出，显示了在区域 0 中有一个很大数量的 SPF 计算。

例 9-264 确定 SPF 多久运行一次

```
R1#show ip ospf
Routing Process "ospf 1" with ID 192.168.254.13
Supports only single TOS(TOS0) routes
It is an area border
SPF schedule delay 5 secs, Hold time between two SPF's 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 8. Checksum Sum 0x48C3E
Number of DCbitless external LSA 0
Number of DoNotAge external LSA 0
Number of areas in this router is 3. 2 normal 1 stub 0 nssa
Area BACKBONE(0)
Number of interfaces in this area is 1
Area has no authentication
SPF algorithm executed 2668 times
```

找出哪个特定的 LSA 在抖动的最简单的办法是打开 **debug ip ospf monitor** 命令。这个 **debug** 确切显示了哪个 LSA 正在抖动。例 9-265 显示了 **debug ip ospf monitor** 的输出并且揭示了一个路由器 LSA 在区域 0 中抖动。

例 9-265 debug ip ospf monitor 的输出指出路由抖动

```
R1# debug ip ospf monitor
OSPF: Schedule SPF in area 0.0.0.0
Change in LS ID 192.168.1.129, LSA type R,
OSPF: schedule SPF: spf_time 1620348064ms wait_interval 10s
```

下一步是到路由器 LSA 抖动的路由器上检查日志中的任何接口抖动。例 9-266 显示了路由器 ID 为 192.168.1.129 的路由器的日志。日志显示一条串行链路持续地启动和关闭。每当有一个接口抖动，就会引发 SPF 的计算。

例 9-266 路由器日志指出引起路由抖动的接口

```
R3#show log
*Mar 29 01:59:07: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed
state to down
*Mar 29 01:59:09: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed
state to up
*Mar 29 01:59:30: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed
state to down
*Mar 29 02:00:03: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed
state to up
```

(2) 解决方法

在这种情况下存在两种实际的解决方法：

- 解决链路抖动问题。
- 重定义区域边界。

有时，第一种解决方法可能不可控制，因为链路抖动是某些电信储运损耗的结果，这超出你的控制范围了，一个暂时的解决办法是手动关闭那个接口。

第二种方法需要进行一些重新设计。如果链路抖动发生太频繁了，就有可能需要重定义区域，将这个路由器排除在区域之外，并使它成为完全端区域中的一员。有时，这也很难实现。

简而言之，链路抖动是现实情况；如果有太多的链路抖动的话，就应该减少区域中的路由器数量以使得更少的路由器受到抖动的影响。

9.9.2 SPF 经常运行——原因：在网络中存在邻居抖动

邻居的抖动也会引起 SPF 计算。邻居可能由于本章中已经讨论过的多个原因而发生抖动。当一条链路关闭时，邻居也同时关闭了。当一个邻居关闭时，它在网络拓扑中引起了一个变化，因此 SPF 开始运行。在图 9-101 中，R3 受困于邻居抖动，并且在区域 0 中的所有路由器都因为这个原因而运行 SPF。

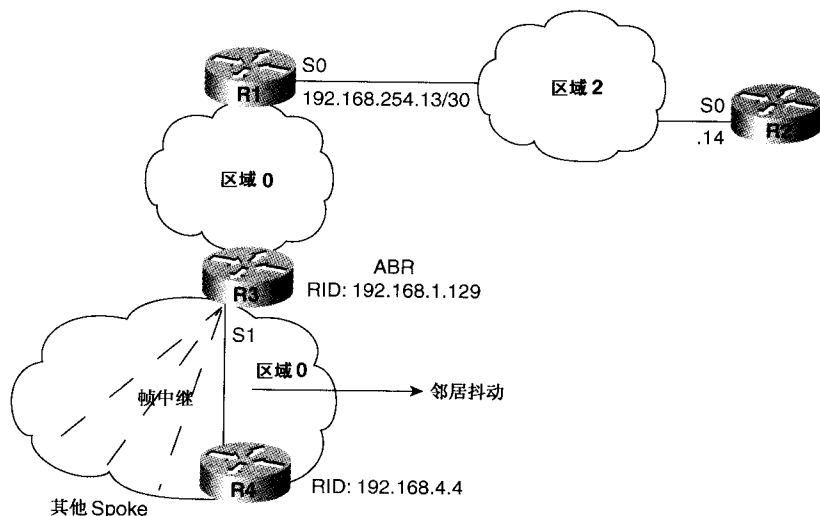


图 9-101 OSPF 邻居抖动引起 SPF 运行

图 9-102 显示了解决这个问题的流程图。

(1) 调试与验证

例 9-267 显示了在区域 0 中 SPF 经常运行。

例 9-267 确定 SPF 多久运行一次

```
R1#show ip ospf
Routing Process "ospf 1" with ID 192.168.254.13
Supports only single TOS(TOS0) routes
It is an area border
SPF schedule delay 5 secs, Hold time between two SPF's 10 secs
```

(待续)

```

Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 8. Checksum Sum 0x48C3E
Number of DCbitless external LSA 0
Number of DoNotAge external LSA 0
Number of areas in this router is 3. 2 normal 1 stub 0 nssa
  Area BACKBONE(0)
    Number of interfaces in this area is 1
    Area has no authentication
    SPF algorithm executed 2458 times

```

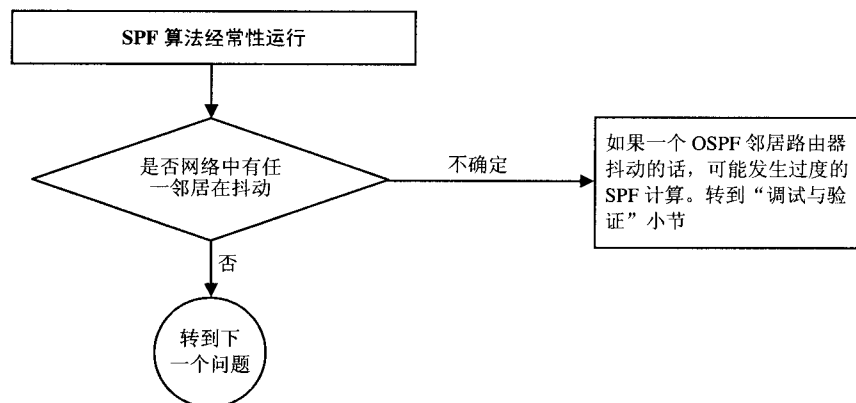


图 9-102 解决问题流程图

下一步要做的事情, 是到 R3 去检查它的日志, 就像在前面例子中做的一样。有一个办法在 OSPF 中跟踪邻居的变化。在 **router ospf** 下配置 **ospf log-adjacency-changes** 来跟踪所有邻居的变化。例 9-268 显示了如何配置 **ospf log-adjacency-changes** 命令。

例 9-268 在 R3 上配置 ospf log-adjacency-changes 命令

```

R3#
router ospf 1
ospf log-adjacency-changes

```

当配置了这条命令之后, 它将在路由器系统日志中保存所有邻居状态的变化。例 9-269 显示了一个 R3 的系统日志消息, 其中显示了邻居状态的改变。输出显示了一个实例, 但是存在许多邻居变化的实例。

例 9-269 R3 的系统日志消息显示 OSPF 状态改变

```

R3#show log

%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from FULL to DOWN, Neighbor Down
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from FULL to INIT, 1-Way
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from Down to INIT, Received Hello
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from INIT to 2WAY, 2-Way Received
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from 2WAY to EXSTART, AdjOK?
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from EXSTART to EXCHANGE

```

(待续)

(待续)

```

Negotiation Done
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from EXCHANGE to LOADING
Exchange Done
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.4 on Serial1 from LOADING to FULL, Loading
Done

```

在某些老版本的 Cisco IOS 软件中不提供 **ospf log-adjacency-changes** 命令或者这个命令不能配置到路由器上。在这种情况下, **show ip ospf neighbor** 命令将会有帮助。例 9-270 显示了 R3 通过 **show ip ospf neighbor** 命令看到了 R4 从 FULL 状态转到 INIT 状态, 接着又转回到 FULL 状态。这个过程持续循环。

例 9-270 确定邻居状态

```

R3#show ip ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
192.168.4.4 1 FULL/- 00:00:34 131.108.1.1 Serial1.1

R2#show ip ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
192.168.4.4 1 INIT/- 00:00:33 131.108.1.1 Serial1.1

R2#show ip ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
192.168.4.4 1 FULL/- 00:00:37 131.108.1.1 Serial1.1

```

(2) 解决方法

这个问题在帧中继星型 (hub-and-spoke) 环境中很常见。如果在帧中继中有太多邻居的话, 它们的 Hello 分组开始被丢弃的机会很高。在这种情况下的解决方法是调整广播队列使得它不再丢弃 OSPF Hello 分组。邻居在 FULL 状态后进入 INIT 状态, 因为邻居缺少 3 个 Hello 分组并且宣布 R2 死亡了。这可以通过看表明串行接口广播队列丢弃了多少分组的 **show interface** 命令的统计数据来确定。例 9-271 显示了 Serial1 接口上 **show interface** 的输出, 其中显示了在帧中继广播队列中有一个很大数量的分组被丢弃。

例 9-271 显示广播队列状态

```

R3#show interface Serial1
Serial1 is up, line protocol is up
Hardware is MK5025
Description: Charlotte Frame Relay Port DLCI 100
MTU 1500 bytes, BW 1024 Kbit, DLY 20000 usec, rely 255/255, load 44/255
Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
LMI enq sent 7940, LMI stat recvd 7937, LMI upd recvd 0, DTE LMI up
LMI enq recvd 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 1023 LMI type is CISCO frame relay DTE
Broadcast queue 64/64, broadcasts sent/dropped 1769202/1849660, interface broadcasts
3579215

```

在例 9-271 中的输出进一步证明了在接口一级存在某些问题。在接口这一级发生了太多分组被丢弃的现象。这引起了路由抖动。要解决这个问题, 你必须相应地调整帧中继广播队

列。调整帧中继广播队列已经超出了本书的范围，但是在 Cisco 站点上有多篇论文讨论了如何调整帧中继广播队列。为进一步研究，可以在如下 URL 参考这些论文：

www.cisco.com/warp/partner/synchronicd/cc/techno/media/wan/frame/prodlit/256_pb.htm
www.cisco.com/warp/public/125/20.html

例 9-272 显示了在解决这个接口丢失分组问题之后，路由抖动现象消失了。广播队列大小由 64 变成了 256。在读完前面提到的调整广播队列的 URL 之后就可以确定正确的数目了。

例 9-272 确认广播队列已经被修复了

```
R3#show interface Serial11
Serial11 is up, line protocol is up
Hardware is MK5025
Description: Charlotte Frame Relay Port DLCI 100
MTU 1500 bytes, BW 1024 Kbit, DLY 20000 usec, rely 255/255, load 44/255
Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
LMI enq sent 7940, LMI stat recvd 7937, LMI upd recvd 0, DTE LMI up
LMI enq recvd 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 1023 LMI type is CISCO frame relay DTE
Broadcast queue 0/256, broadcasts sent/dropped 1769202/0 , interface broadcasts
3579215
```

9.9.3 SPF 经常运行——原因：路由器 ID 重复了

在 OSPF 中这是一个常见问题。当两个路由器有一样的路由器 ID 的时候，就会在 OSPF 拓扑数据库中发生混乱，并且路由持续地被增或者被删。这个问题最常见的征兆就是 LS Age 字段总是一个很小的值。

这个问题通常是由将一个路由器的配置复制并粘贴到另一个路由器中这种行为引起的。这导致了两个路由器有同样的路由器 ID。图 9-103 显示了一个网络设置，其中 R2 和 R3 有重复的路由器 ID：192.168.1.129。

例 9-103 OSPF 网络中有重复路由器 ID

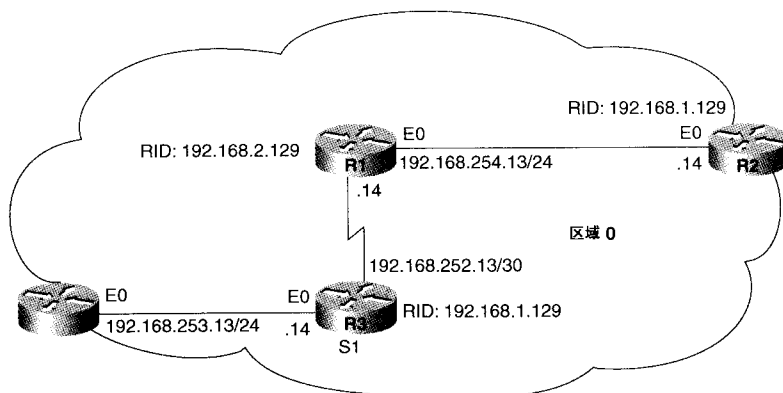


图 9-103 显示了解决这个问题应遵循的流程图

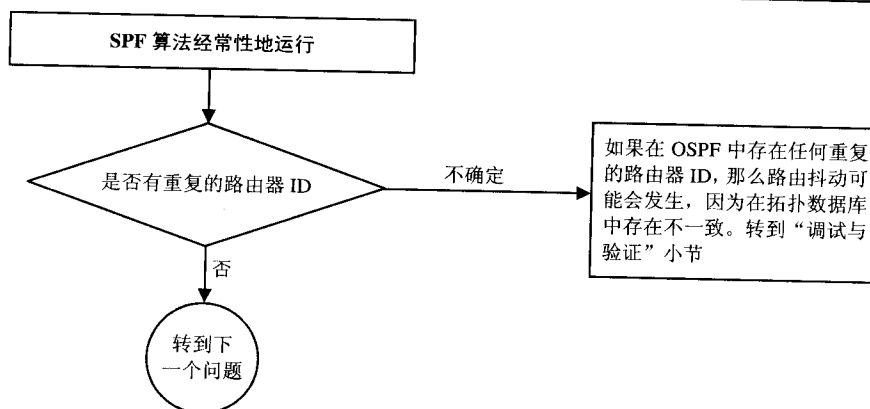


图 9-104 解决问题流程图

(1) 调试与验证

当有重复的路由器 ID 的时候, 它将会频繁地引起 SPF 计算, 并且 SPF 计数器持续增加, 直到这个问题被解决了为止。例 9-273 显示了在区域 0 中的 SPF 运行了 2446 次, 这是个很大的数目。

例 9-273 确定 SPF 多久运行一次

```

R1#show ip ospf
Routing Process "ospf 1" with ID 192.168.2.129
Supports only single TOS(TOS0) routes
It is an area border
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 8. Checksum Sum 0x48C3E
Number of DCbitless external LSA 0
Number of DoNotAge external LSA 0
Number of areas in this router is 4. 1 normal 0 stub 0 nssa
Area BACKBONE(0)
  Number of interfaces in this area is 1
  Area has no authentication
  SPF algorithm executed 2446 times
  
```

下一步是打开 **debug ip ospf monitor** 命令。这个调试确切显示跟踪哪个 LSA。例 9-274 显示了 **debug ip ospf monitor** 的输出, 表明一个路由器 ID 为 192.168.1.129 的路由器就是问题所在。输出也显示那是一个路由器 LSA。

例 9-274 debug ip ospf monitor 输出指出了引起这个问题的路由器

```

R1# debug ip ospf monitor
OSPF: Schedule SPF in area 0.0.0.0
Change in LS ID 192.168.1.129, LSA type R,
OSPF: schedule SPF: spf_time 1620348064ms wait_interval 10s
  
```

例 9-275 显示了有问题的路由器 LSA 的输出。这个输出的两个实例分别持续 15 秒钟。第一个输出显示这个路由器上链路的数目是 1; 第 2 个输出显示在这个路由器链路的数目是 3。

这是由于存在重复路由器 ID 而引起的差异。这表示一定存在另一个具有同样路由器 ID 的路由器，它导致了链路的数量每 15 秒变化一次。LS Age 字段也总是小于 10 秒。

这个例子中的第一个输出是 R2 的路由器 LSA；第二个输出是 R3 的路由器 LSA。

例 9-275 确定路由器 LSA 的差异

```
R1#show ip ospf database router 192.168.1.129

OSPF Router with ID (192.168.2.129) (Process ID 1)

Router Link States (Area 0.0.0.0)

LS age: 9
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 192.168.1.129
Advertising Router: 192.168.1.129
LS Seq Number: 80067682
Checksum: 0xC456
Length: 36
Number of Links: 1

Link connected to: a Transit Network
(Link ID) Designated Router address: 192.168.254.14
(Link Data) Router Interface address: 192.168.254.14
Number of TOS metrics: 0
TOS 0 Metrics: 10

R1#show ip ospf database router 192.168.1.129

OSPF Router with ID (192.168.2.129) (Process ID 1)

Router Link States (Area 0.0.0.0)

LS age: 7
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 192.168.1.129
Advertising Router: 192.168.1.129
LS Seq Number: 80067683
Checksum: 0xA7D8
Length: 60
Number of Links: 3

Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 192.168.2.129
(Link Data) Router Interface address: 192.168.252.13
Number of TOS metrics: 0
TOS 0 Metrics: 66

Link connected to: a Stub Network
(Link ID) Network/subnet number: 192.168.252.12
(Link Data) Network Mask: 255.255.255.252
```

(待续)

重复
动可
据库
试与

计数器持续增加，
次，这是个很大

LSA。例 9-274
的路由器就是问

制持续 15 秒钟。
链路的数目是 3。

```

Number of TOS metrics: 0
TOS 0 Metrics: 66

Link connected to: a Transit Network
(Link ID) Designated Router address: 192.168.253.14
(Link Data) Router Interface address: 192.168.253.14
Number of TOS metrics: 0
TOS 0 Metrics: 1

```

R1#

例 9-276 显示了 R2 和 R3 具有相同的路由器 ID。

例 9-276 确定重复的路由器 ID

```

R2#show ip ospf
Routing Process "ospf 1" with ID 192.168.1.129

R3#show ip ospf
Routing Process "ospf 1" with ID 192.168.1.129

```

(2) 解决方法

要解决这个问题，要么改变 R3 的路由器 ID，要么改变 R2 的路由器 ID。例 9-277 显示了如何变 R3 的路由器 ID 并给出了 **show ip ospf** 命令的输出来确认路由器 ID 已经改变了。

例 9-277 改变 R3 的路由器 ID

```

R3(config)#interface loopback 0
R3(config-if)#ip address 192.168.3.129 255.255.255.255
R3(config-if)#end

R3#show ip ospf
Routing Process "ospf 1" with ID 192.168.3.129

```

例 9-278 显示了在改变 R3 的路由器 ID 之后，192.168.1.129 的 LS 年龄就在 OSPF 数据库中变得稳定了。LS 年龄达到了 90 秒，因此入口现在稳定了。

例 9-278 问题 LSA 的 LS 年龄现在稳定了

```

R1#show ip ospf database router 192.168.1.129

      OSPF Router with ID (192.168.2.129) (Process ID 1)

          Router Link States (Area 0.0.0.0)

LS age: 90
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 192.168.1.129
Advertising Router: 192.168.1.129
LS Seq Number: 80067686

```

(待续)

```
Checksum: 0xC456
Length: 36
Number of Links: 1

Link connected to: a Transit Network
(Link ID) Designated Router address: 192.168.254.14
(Link Data) Router Interface address: 192.168.254.14
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

9.9.4 常见 OSPF 出错消息

本节讨论在 OSPF 中常见的错误消息。一些消息是错误的征兆，但这些消息不在这一节中讨论。一些消息其义自明，如同下面这个：

```
Warning: Router is currently an ASBR while having only one area which is a stub area
```

这条警告消息表示你正试图向一个端区域重发布路由。

下面是将在本节中讨论的出错消息列表：

- “Unknown routing protocol”（未知的路由协议）。
- “OSPF: Could not allocate router id”（OSPF：不能分配路由器 ID）
- “%OSPF-4-BADLSATYPE”。
- “%OSPF-4-ERRRCV”。

9.9.5 “Unknown routing Protocol” 出错消息

当在一个路由器上键入命令 `router ospf 1` 来配置 OSPF 的时候会产生这个错误消息。这个消息表示软件或者硬件不支持 OSPF。通常的低端平台，如 1000 和 1600 系列的路由器，需要一个特别的内核映象（image）（也就是，附加的功能集）来运行 OSPF。另一些低端平台，如 800 系列路由器，不支持 OSPF。

9.9.6 “OSPF: Could not allocate routerid” 出错消息

这个消息出现在两种情况下：

- 没有具有可用 IP 地址的 up/up 接口。
- 对多个 OSPF 进程没有足够的具有可用 IP 地址的 UP 接口。

OSPF 需要一个有效的 IP 地址，其状态是 up/up 状态，这样它就可以为 OSPF 进程分配路由 ID 了。这个 IP 地址必须被分配到一个 up/up 接口上。如果一个路由器分配路由器 ID 失败了，OSPF 将不工作。这个问题可以通过使用回环地址来解决。

回环接口的解决方法在两种情况下都有用。为一个进程只要配置一个回环接口。如果你试图运行多个进程的话，就需要多个回环接口。

9-277 显示
经改变了。

OSPF 数据

（待续）

9.9.7 “%OSPF-4-BADLSATYPE: Invalid Isa: Bad LSA type” 类型 6 出错消息

如果邻居路由器正发送组播 OSPF (MOSPF) 分组的时候出现这种问题很常见。关于 MOSPF 的更多信息, 请参考 RFC1584。Cisco 路由不支持 MOSPF, 因此它们简单地忽略它。要去掉这些消息, 简单地键入如下命令:

```
router ospf 1
  ignore lsa mospf
```

如果出错类型不是 6 的话, 那么它可能是一个 bug 或者是一个内存崩溃错误。参考第 9.8 节 “OSPF 邻居停滞于 LOADING 状态” 了解更多的关于如何解决 BAD LSA 问题的内容。

9.9.8 “OSPF-4-ERRRCV” 出错消息

这条消息表示 OSPF 收到了无效的分组。

可能产生 3 种常见类型的这种消息:

- 不匹配的区域 ID。
- 校验和出错。
- 在接收方接口没有启用 OSPF。

1. 不匹配的区域 ID

这条消息看起来像这样:

```
%OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be
virtual-link but not found from 170.170.3.3, Ethernet0
```

这表示与这个接口相连的邻居的接口在区域 0 中, 但是这个接口本身不在区域 0 中。在这种情况下, 路由器将不与发送这个分组的路由器形成 OSPF 邻接体关系。如果某一边的虚链路配错了的话这种现象也会发生。

为了避免出现这些消息, 通过检查在路由器配置中 OSPF 语句下的 network 语句来确信两边都有一样的区域 ID。例如, 如果在两个路由器之间的链路 10.10.10.0/24 在区域 1 中, 则确信在两边路由器上的 network 语句包括这条特殊的链路到区域 1 中。

network 命令看起来像这样:

```
router ospf 1
  network 10.10.10.0 0.0.0.255 area 1
```

如果配置了虚链路, 那么重复地检查虚链路的配置。

2. 校验和出错

这个消息看起来像这样:

```
%OSPF-4-ERRRCV: Received invalid packet: Bad Checksum from 144.100.21.141,
TokenRing0/0
```

这表示 OSPF 遇到了一个所收到的分组中的错误。这是由于 OSPF 校验和与这个路由器收到的 OSPF 分组不匹配。

这个问题有 3 个原因：

- 在邻居中间的一个设备，例如交换机，破坏了分组。
- 发送方路由器发送的分组不可用。在这种情况下，要么是发送方路由器的接口损坏了，要么是一个软件 bug 引起了这个错误。
- 接收方路由器计算出了错误的校验和。在这种情况下，要么是接收方路由器的接口损坏了要么是一个软件 bug 引起了这个错误。这是这条错误消息最不可能发生的原因。

这个问题很难排解，但是你可以从下面这些解决方法开始，这些办法在 90% 的这类情况中都是有效的。依照这些步骤一步步来是很重要的：

步骤 1：换掉路由器之间的电缆。对于本节中给出的例子来说，这两个路由器一个是发送错误分组的路由器（144.100.21.141）而另一个是对这些错误分组报错的路由器。

步骤 2：如果步骤 1 不能解决问题，那么在路由器之间的交换机上使用一个不同的端口。

步骤 3：如果步骤 2 不能解决问题，那么用一条交叉电缆（cross-over cable）直接连接两个路由器。如果收不到进一步的消息，那么很有可能是交换机破坏了分组。

如果上面这些步骤都不能解决问题，那么与 Cisco 技术支持中心（TAC）联系并且与一个工程师一道工作来找出在 Cisco IOS 软件中的一个 bug 或者为部分的或者全部的部件替代得到一个可能的返回材料授权（RMA）。

3. 在接收方接口没有启用 OSPF

这条消息看起来像这样：

```
%OSPF-4-ERRRCV: Received invalid packet: OSPF not enabled on interface from
141.108.16.4, Serial0.100
```

产生这条消息的路由器在 Serial0.100 接口接收到一个从 141.108.16.4 发送过来的分组，但是在 Serial0.100 接口没有启用 OSPF，这条消息仅仅为一个非 OSPF 接口产生一次。