

Por que o OSPF não forma adjacência em um PRI, BRI ou interface do discador?

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Conventions](#)

[O problema](#)

[A solução](#)

[Informações Relacionadas](#)

[Introduction](#)

Esta Nota Técnica explica um problema com a formação da adjacência OSPF quando as interfaces do discador são configuradas como links ponto-a-ponto.

[Prerequisites](#)

[Requirements](#)

Não existem requisitos específicos para este documento.

[Componentes Utilizados](#)

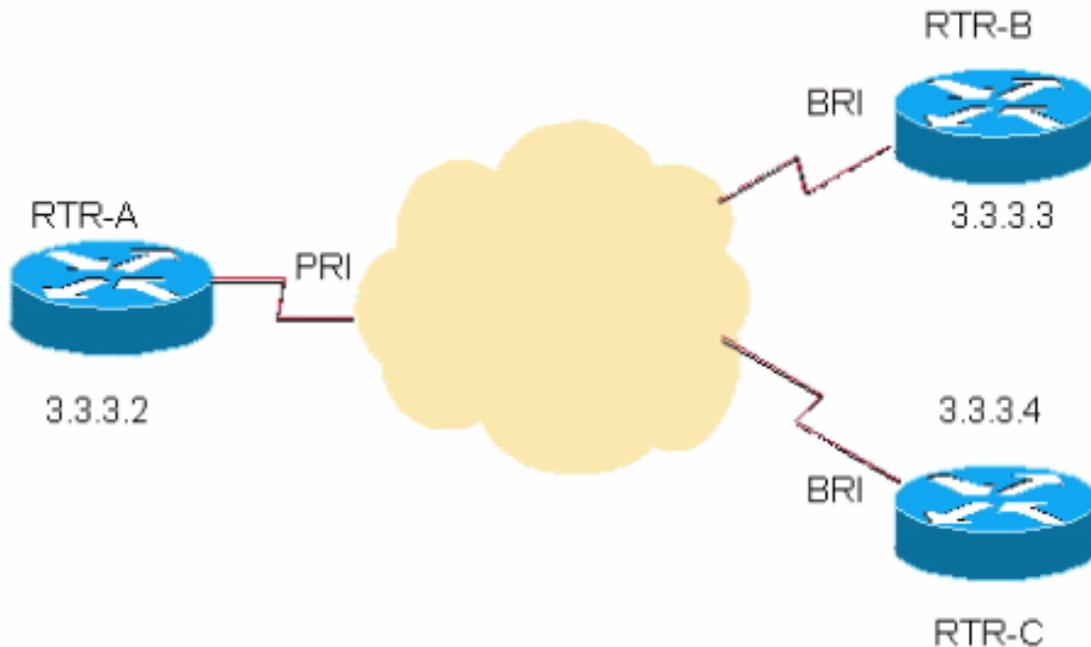
Este documento não se restringe a versões de software e hardware específicas.

[Conventions](#)

For more information on document conventions, refer to the [Cisco Technical Tips Conventions](#).

[O problema](#)

O tipo de rede OSPF nas interfaces PRI (Primary Rate Interface Interface de Taxa Primária), BRI (Basic Rate Interface) e de discador é ponto-a-ponto, o que significa que uma interface não pode formar adjacência com mais de um vizinho. Um problema comum quando uma interface PRI, BRI ou de discador tenta formar uma adjacência de OSPF é que o vizinho fica preso no processo de exstart/exchange. Vamos ver um exemplo.



Usando o comando **show ip ospf neighbor**, podemos ver que o estado do vizinho está preso em "EXSTART".

RTR-A# **show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
3.3.3.3	1	EXSTART/ -	00:00:37	3.3.3.3	Serial6/0:23
3.3.3.4	1	EXSTART/ -	00:00:39	3.3.3.4	Serial6/0:23

RTR-B# **show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
3.3.3.2	1	EXSTART/ -	00:00:36	3.3.3.2	BRI0

RTR-C# **show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
3.3.3.2	1	EXSTART/ -	00:00:35	3.3.3.2	BRI0

A configuração do RTR-Bs mostra que o tipo de rede é ponto-a-ponto:

RTR-B# **show ip ospf interface bri0**

```

BRI0 is up, line protocol is up (spoofing)
  Internet Address 3.3.3.3/24, Area 2
  Process ID 1, Router ID 3.3.3.3, 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)

```

Podemos depurar essa situação usando o comando **debug ip ospf adj**. Vamos ver alguns exemplos de saída obtidos ao executar este comando no RTR-B na figura acima:

- 1: Send DBD to 3.3.3.2 on BRI0 seq 0xB41 opt 0x42 flag 0x7 len 32
- 2: Rcv DBD from 3.3.3.2 on BRI0 seq 0x1D06 opt 0x42 flag 0x7 len 32

```
mtu 1500 state EXSTART
3: First DBD and we are not SLAVE
4: Rcv DBD from 3.3.3.2 on BRI0 seq 0xB41 opt 0x42 flag 0x2 len 92 mtu
  1500 state EXSTART
5: NBR Negotiation Done. We are the MASTER
6: Send DBD to 3.3.3.2 on BRI0 seq 0xB42 opt 0x42 flag 0x3 len 92
7: Database request to 3.3.3.2
8: sent LS REQ packet to 3.3.3.2, length 12
9: Rcv DBD from 3.3.3.2 on BRI0 seq 0x250 opt 0x42 flag 0x7 len 32
  mtu 1500 state EXCHANGE
10: EXCHANGE - inconsistent in MASTER/SLAVE
11: Bad seq received from 3.3.3.2 on BRI0
12: Send DBD to 3.3.3.2 on BRI0 seq 0x2441 opt 0x42 flag 0x7 len 32
13: Rcv DBD from 3.3.3.2 on BRI0 seq 0x152C opt 0x42 flag 0x2 len 92
  mtu 1500 state EXSTART
14: Unrecognized dbd for EXSTART
15: Rcv DBD from 3.3.3.2 on BRI0 seq 0xB42 opt 0x42 flag 0x0 len 32
  mtu 1500 state EXSTART
16: Unrecognized dbd for EXSTART
```

Linhas 1 a 3: O RTR-B envia o primeiro DBD para 3.3.3.2 (RTR-A) com seq 0xB41 e recebe o primeiro DBD de 3.3.3.2 (RTR-A) com seq# 0x1D06. A negociação de vizinhos ainda não está concluída.

Linhas 4 a 6: O RTR-B recebe uma resposta de 3.3.3.2 (RTR-A) indicando que o RTR-A recebeu o primeiro DBD do RTR-B. Como o RTR-B tem o maior ID de roteador, o RTR-A se elege escravo. Depois de receber a confirmação do RTR-A, o RTR-B se declara mestre e envia o primeiro DBD com dados nele contidos. Observe o número de sequência, que é 0xB42. Como o RTR-B é o mestre, somente ele pode incrementar o número de sequência.

Linha 7: O RTR-B solicita dados do RTR-A já que o RTR-A indicou que tem mais dados para enviar (sinalizador definido como 0x2 no último DBD recebido do RTR-A).

Linha 8: O RTR-B envia um pacote de solicitação de link-state para 3.3.3.2 (RTR-A). Este é um pacote OSPF tipo 3. Geralmente, esse pacote é enviado ao endereço IP do vizinho. Nesse caso, o endereço IP do vizinho é o ID do roteador.

Linhas 9 a 11: O RTR-B recebe uma resposta do escravo (RTR-A) com um número de sequência completamente diferente e um flag de 0x7, que é o flag de inicialização. Esse DBD foi projetado para outro roteador (provavelmente RTR-C), mas o RTR-B o recebeu incorretamente. O RTR-B declara que há uma discrepância porque um sinalizador de 0x7 significa que o escravo alterou seu status para mestre definindo o bit MS (Mestre/Escravo) durante a troca de adjacência. O RTR-B também reclama do número de sequência porque está fora de ordem. O escravo deve sempre seguir o número de sequência do mestre.

Linha 12: O RTR-B reinicializa a adjacência enviando o primeiro DBD para 3.3.3.2 para reeleger mestre e escravo.

Linhas 13 a 14: O RTR-B recebe um DBD de 3.3.3.2 (RTR-A), indicando que é um escravo, sem reconhecer o número de sequência do RTR-B. O RTR-B declara que não reconhece esse DBD, já que a negociação principal e escrava ainda não foi concluída. Este pacote DBD foi destinado a outro roteador.

Linha 15: O RTR-B recebe uma resposta de 3.3.3.2 (RTR-A) para o DBD antigo, mas é muito tarde porque o RTR-B já reinicializou o processo de adjacência.

Linha 16: O RTR-B não reconhece esse DBD porque ele é para uma adjacência "antiga", que o

RTR-B já destruiu.

Esse processo será repetido infinitamente.

A solução

De acordo com a seção 8.1 do [RFC 2328](#), o OSPF envia um pacote multicast para um tipo de rede ponto-a-ponto mesmo depois que a interface atinge o estado bidirecional. Como o RTR-A está tentando formar adjacências com o RTR-B e o RTR-C, o RTR-B recebe pacotes DBD destinados ao RTR-C e o RTR-C recebe pacotes DBD destinados ao RTR-B.

Para resolver esse problema, altere o tipo de rede em todos os roteadores para point-to-multipoint. Isso altera o comportamento do OSPF para enviar pacotes unicast após o estado de 2 vias. Agora o RTR-B recebe apenas pacotes destinados a si mesmo e o RTR-C recebe pacotes destinados a si mesmo. Alterar o tipo de rede dessa forma garante que o roteador OSPF formará adjacência em uma interface PRI, BRI ou de discador.

Para alterar o tipo de rede, insira os seguintes comandos de configuração, terminando cada linha pressionando ENTER. Vamos alterar o RTR-B como exemplo.

```
RTR-B# configure terminal
RTR-B(config)# int bri 0
RTR-B(config-if)# ip ospf network point-to-multipoint
RTR-B(config-if)# end
```

Agora, se observarmos os comandos **show** para RTR-B, podemos verificar se o tipo de rede é ponto-a-multiponto e se o estado está cheio.

```
RTR-B# show ip ospf interface bri0
BRI0 is up, line protocol is up (spoofing)
  Internet Address 3.3.3.3/24, Area 2
  Process ID 1, Router ID 3.3.3.3, 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 00:00:16
  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 1
    Adjacent with neighbor 172.16.141.10
  Suppress hello for 0 neighbor(s)
```

```
RTR-B# show ip ospf neighbor
Neighbor ID      Pri   State           Dead Time   Address        Interface
172.16.141.10    1    FULL/  -        00:01:36    3.3.3.2       BRI0
```

Informações Relacionadas

- [Configurando o Dialup BRI-to-BRI com os mapas de discadores DDR](#)
- [Página de suporte de OSPF](#)
- [Suporte Técnico - Cisco Systems](#)