

Por que os vizinhos de OSPF estão presos no estado Exstart/Exchange?

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Estado de Exstart](#)

[Estado de intercâmbio](#)

[Vizinhos presos no estado Exstart/Exchange](#)

[A solução](#)

[Informações Relacionadas](#)

[Introdução](#)

Os estados OSPF para a formação de adjacência são Down, Init, Attempt, 2-way, Exstart, Exchange, Loading e Full. Pode haver um número de motivos pelos quais os vizinhos Open Shortest Path First (OSPF) estejam presos em estado exstart/exchange. Este documento concentra-se em uma compatibilidade da MTU entre os vizinhos OSPF, resultando em um estado exstart/exchange. [Para obter mais detalhes sobre o troubleshooting do OSPF, consulte Troubleshooting OSPF.](#)

[Pré-requisitos](#)

[Requisitos](#)

Os leitores deste documento devem ser familiares com

[Componentes Utilizados](#)

As informações neste documento são baseadas nestas versões de software e hardware:

- Cisco 2503 Routers
- Liberação do Cisco IOS ® Software 12.2(24a) que é executado em ambos os roteadores

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto potencial de qualquer comando.

Convenções

Para obter mais informações sobre convenções de documento, consulte as [Convenções de dicas técnicas Cisco](#).

Estado de Exstart

Depois de dois roteadores vizinhos de OSPF estabelecerem comunicação bidirecional e concluírem a eleição VR/BDR (em redes multiacesso), os roteadores fazem transição para o estado exstart. Neste estado, os roteadores vizinho estabelecem um mestre/relação escrava e determinam o número de sequência inicial do Database Descriptor (DBD) usar-se ao trocar pacotes DBD.

Estado de intercâmbio

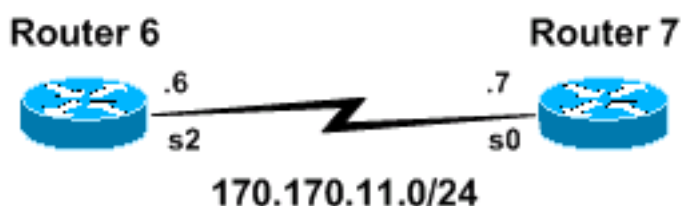
Uma vez que o mestre/relação escrava esteve negociado (o roteador com o Roteador-ID o mais alto se transforma o mestre), a transição dos roteadores vizinho no estado de intercâmbio. Nesse estado, os roteadores trocam pacotes de DBD, que descrevem todo o banco de dados do estado de link. Os roteadores também enviam pacotes de solicitação de estado de enlace, que solicitam mais Anúncios de estado de enlace (LSA) dos vizinhos.

Embora a transição dos vizinhos de OSPF através dos estados de intercâmbio/exstart durante o processo de construção de contiguidade normal OSPF, ele não seja normal para que os vizinhos de OSPF sejam colados neste estado. O motivo a seguir é o mais comum para a estabilização de vizinhos OSPF nesse estado. Refira [estados do vizinho OSPF](#) para aprender mais sobre os estados diferentes OSPF.

Vizinhos presos no estado Exstart/Exchange

O problema ocorre mais frequentemente ao tentar executar o OSPF entre um roteador Cisco e o roteador de um outro vendedor. O problema ocorre quando os ajustes da unidade de transmissão máxima (MTU) para relações do roteador vizinho não combinam. Se o roteador com o MTU mais alto envia um pacote maior que o MTU ajustado no roteador vizinho, o roteador vizinho ignore o packet.0 quando este problema ocorre, a saída do comando `show ip ospf neighbor` indica a saída similar isso mostrado [abaixo](#).

Uma recriação real deste problema é descrita nesta seção.



O roteador 6 e o roteador 7 na topologia acima são conectados através do Frame Relay e o roteador 6 foi configurado com as rotas estáticas 5 redistribuídas no OSPF. A interface serial no roteador 6 tem o MTU padrão de 1500, quando a interface serial no roteador 7 tiver um MTU de

1450. Cada configuração de roteador é mostrada na tabela abaixo (somente a informação de configuração necessária é mostrada):

Configuração do Roteador 6	Configuração do Roteador 7
<pre>interface Serial2 !--- MTU sets to it's default value of 1500. no ip address no ip directed-broadcast encapsulation frame-relay no ip mroute-cache frame- relay lmi-type ansi ! interface Serial2.7 point-to-point ip address 170.170.11.6 255.255.255.0 no ip directed-broadcast frame-relay interface-dlci 101 ! router ospf 7 redistribute static subnets network 170.170.11.0 0.0.0.255 area 0 ip route 192.79.34.0 255.255.255.0 Null0 ip route 192.79.35.0 255.255.255.0 Null0 ip route 192.79.36.0 255.255.255.0 Null0 ip route 192.79.37.0 255.255.255.0 Null0 ip route 192.79.38.0 255.255.255.0 Null0</pre>	<pre>! interface Serial0 mtu 1450 no ip address no ip directed- broadcast encapsulation frame-relay frame-relay lmi- type ANSI ! interface Serial0.6 point- to-point ip address 170.170.11.7 255.255.255.0 no ip directed- broadcast frame-relay interface-dlci 110! ! router ospf 7 network 170.170.11.0 0.0.0.255 area 0</pre>

O comando `show ip ospf neighbor` output para cada roteador é:

```
router-6# show ip ospf neighbor Neighbor ID Pri State Dead Time Address Interface 170.170.11.7 1
EXCHANGE/ - 00:00:36 170.170.11.7 Serial2.7 router-6# router-7# show ip ospf neighbor Neighbor
ID Pri State Dead Time Address Interface 170.170.11.6 1 EXSTART/ - 00:00:33 170.170.11.6
Serial0.6 router-7#
```

O problema ocorre quando o roteador 6 envia um pacote DBD com mais de 1450 bytes (MTU do roteador 7) enquanto está no estado de intercâmbio. Use o pacote **debugger IP** e os comandos **debug ip ospf adj** em cada roteador ver o processo da adjacência de OSPF como ocorre. A saída do roteador 6 e 7 de etapas 1 a 4 é:

1. Resultado do debug do roteador 6: `***ROUTER6 IS SENDING HELLOS BUT HEARS NOTHING, STATE OF NEIGHBOR IS DOWN`
`00:03:53: OSPF: 170.170.11.7 address 170.170.11.7 on Serial2.7 is dead`
`00:03:53: OSPF: 170.170.11.7 address 170.170.11.7 on Serial2.7 is dead, state DOWN`
2. Resultado do debug do roteador 7: `OSPF NOT ENABLED ON ROUTER7 YET`
3. Resultado do debug do roteador 6: `***ROUTER6 SENDING HELLOS`
`00:03:53: IP: s=170.170.11.6 (local), d=224.0.0.5 (Serial2.7), len 64, sending broad/multicast, proto=89`
`00:04:03: IP: s=170.170.11.6 (local), d=224.0.0.5 (Serial2.7), Len 64, sending broad/multicast, proto=89`
4. Resultado do debug do roteador 7: `OSPF NOT ENABLED ON ROUTER7 YET`

5. Resultado do debug do roteador 7:***OSPF ENABLED ON ROUTER7, BEGINS SENDING HELLOS AND BUILDING A ROUTER LSA
00:17:44: IP: s=170.170.11.7 (local), d=224.0.0.5 (Serial0.6), Len 64, sending broad/multicast, proto=89
00:17:44: OSPF: Build router LSA for area 0, router ID 170.170.11.7, seq 0x80000001
6. Resultado do debug do roteador 6:***RECEIVE HELLO FROM ROUTER7
00:04:04: IP: s=170.170.11.7 (Serial2.7), d=224.0.0.5, Len 64, rcvd 0, proto=89
00:04:04: OSPF: Rcv hello from 170.170.11.7 area 0 from Serial2.7 170.170.11.7
00:04:04: OSPF: End of hello processing
7. Resultado do debug do roteador 6:***ROUTER6 SEND HELLO WITH ROUTER7 ROUTERID IN THE HELLO PACKET
00:04:13: IP: s=170.170.11.6 (local), d=224.0.0.5 (Serial2.7), Len 68, sending broad/multicast, proto=89
8. Resultado do debug do roteador 7:***ROUTER7 RECEIVES HELLO FROM ROUTER6 CHANGES STATE TO 2WAY
00:17:53: IP: s=170.170.11.6 (Serial0.6), d=224.0.0.5, Len 68, rcvd 0, proto=89
00:17:53: OSPF: Rcv hello from 170.170.11.6 area 0 from Serial0.6 170.170.11.6
00:17:53: OSPF: 2 Way Communication to 170.170.11.6 on Serial0.6, state 2WAY
9. Resultado do debug do roteador 7:***ROUTER7 SENDS INITIAL DBD PACKET WITH SEQ# 0x13FD
00:17:53: OSPF: Send DBD to 170.170.11.6 on Serial0.6 seq 0x13FD opt 0x2 flag 0x7 Len 32
00:17:53: IP: s=170.170.11.7 (local), d=224.0.0.5 (Serial0.6), Len 52, sending broad/multicast, proto=89
00:17:53: OSPF: End of hello processing
10. Resultado do debug do roteador 6:***ROUTER6 RECEIVES ROUTER7'S INITIAL DBD PACKET CHANGES STATE TO 2-WAY
00:04:13: IP: s=170.170.11.7 (Serial2.7), d=224.0.0.5, Len 52, rcvd 0, proto=89
00:04:13: OSPF: Rcv DBD from 170.170.11.7 on Serial2.7 seq 0x13FD opt 0x2 flag 0x7 Len 32 mtu 1450 state INIT
00:04:13: OSPF: 2 Way Communication to 170.170.11.7 on Serial2.7, state 2WAY
11. Resultado do debug do roteador 6:***ROUTER6 SENDS DBD PACKET TO ROUTER7 (MASTER/SLAVE NEGOTIATION - ROUTER6 IS SLAVE)
00:04:13: OSPF: Send DBD to 170.170.11.7 on Serial2.7 seq 0xE44 opt 0x2 flag 0x7 Len 32
00:04:13: IP: s=170.170.11.6 (local), d=224.0.0.5 (Serial2.7), Len 52, sending broad/multicast, proto=89
00:04:13: OSPF: NBR Negotiation Done. We are the SLAVE
12. Resultado do debug do roteador 7:***RECEIVE ROUTER6'S INITIAL DBD PACKET (MTU MISMATCH IS RECOGNIZED)
00:17:53: IP: s=170.170.11.6 (Serial0.6), d=224.0.0.5, Len 52, rcvd 0, proto=89
00:17:53: OSPF: Rcv DBD from 170.170.11.6 on Serial0.6 seq 0xE44 opt 0x2 flag 0x7 Len 32 mtu 1500 state EXSTART
00:17:53: OSPF: Nbr 170.170.11.6 has larger interface MTU
13. Resultado do debug do roteador 6:***SINCE ROUTER6 IS SLAVE SEND DBD PACKET WITH LSA HEADERS, SAME SEQ# (0x13FD) TO ACK ROUTER7'S DBD. (NOTE SIZE OF PKT)
00:04:13: OSPF: Send DBD to 170.170.11.7 on Serial2.7 seq 0x13FD opt 0x2 flag 0x2 Len 1472
00:04:13: IP: s=170.170.11.6 (local), d=224.0.0.5 (Serial2.7), Len 1492, sending broad/multicast, proto=89
14. Resultado do debug do roteador 7:***NEVER RECEIVE ACK TO ROUTER7'S INITIAL DBD,

```
RETRANSMIT
00:17:54: IP: s=170.170.11.7 (local), d=224.0.0.5
(Serial0.6), Len 68, sending broad/multicast, proto=89
00:18:03: OSPF: Send DBD to 170.170.11.6 on Serial0.6
seq 0x13FD opt 0x2 flag 0x7 Len 32 00:18:03: OSPF:
Retransmitting DBD to 170.170.11.6 on Serial0.6 [1]
```

Neste momento, o roteador 6 mantém-se tentar ao ACK o pacote DBD inicial do roteador 7.

```
00:04:13: IP: s=170.170.11.7 (Serial2.7), d=224.0.0.5,
Len 68, rcvd 0, proto=89
00:04:13: OSPF: Rcv hello from 170.170.11.7 area 0 from
Serial2.7 170.170.11.7
00:04:13: OSPF: End of hello processing

00:04:18: IP: s=170.170.11.7 (Serial2.7), d=224.0.0.5,
Len 52, rcvd 0, proto=89
00:04:18: OSPF: Rcv DBD from 170.170.11.7 on Serial2.7
seq 0x13FD opt 0x2 flag 0x7 Len 32 mtu 1450 state EXCHANGE

00:04:18: OSPF: Send DBD to 170.170.11.7 on Serial2.7
seq 0x13FD opt 0x2 flag 0x2 Len 1472
00:04:18: IP: s=170.170.11.6 (local), d=224.0.0.5
(Serial2.7), Len 1492, sending broad/multicast, proto=89

00:04:23: IP: s=170.170.11.6 (local), d=224.0.0.5
(Serial2.7), Len 68, sending broad/multicast, proto=89

00:04:23: IP: s=170.170.11.7 (Serial2.7), d=224.0.0.5,
Len 52, rcvd 0, proto=89
00:04:23: OSPF: Rcv DBD from 170.170.11.7 on Serial2.7
seq 0x13FD opt 0x2 flag 0x7 Len 32 mtu 1450 state EXCHANGE
```

O roteador 7 nunca obtém um ACK do roteador 6 porque o pacote DBD do roteador 7 é demasiado grande para o roteador 7 MTU. O roteador 7 retransmite repetidamente o pacote DBD.

```
0:17:58: IP: s=170.170.11.7 (local), d=224.0.0.5
(Serial0.6), Len 52, sending broad/multicast, proto=89
00:18:03: OSPF: Send DBD to 170.170.11.6 on Serial0.6
seq 0x13FD opt 0x2 flag 0x7 Len 32 00:18:03: OSPF:
Retransmitting DBD to 170.170.11.6 on Serial0.6 [2]

00:18:03: IP: s=170.170.11.7 (local), d=224.0.0.5
(Serial0.6), Len 52, sending broad/multicast, proto=89
00:18:03: IP: s=170.170.11.6 (Serial0.6), d=224.0.0.5,
Len 68, rcvd 0, proto=89
00:18:03: OSPF: Rcv hello from 170.170.11.6 area 0 from
Serial0.6 170.170.11.6
00:18:03: OSPF: End of hello processing

00:18:04: IP: s=170.170.11.7 (local), d=224.0.0.5
(Serial0.6), Len 68, sending broad/multicast, proto=89

00:18:03: OSPF: Send DBD to 170.170.11.6 on Serial0.6
seq 0x13FD opt 0x2 flag 0x7 Len 32 00:18:03: OSPF:
Retransmitting DBD to 170.170.11.6 on Serial0.6 [3]

00:18:08: IP: s=170.170.11.7 (local), d=224.0.0.5
(Serial0.6), Len 52, sending broad/multicast, proto=89
router-7#
```

Porque o roteador 6 tem um MTU mais alto, continua a aceitar o pacote DBD do roteador 7, tenta reconhecê-los, e permanece no estado de intercâmbio.

Porque o roteador 7 tem um MTU inferior, ignora os pacotes DBD junto com o ACK do roteador 6, continua a retransmitir o pacote DBD inicial, e sobras no estado de EXSTART.

Deixe-nos olhar algumas das etapas acima. Em etapa 9 e 11, o roteador 7 e o roteador 6 enviam seus primeiros pacotes DBD com a bandeira 0x7 ajustada como parte da negociação do mestre/escravo. Após a determinação do mestre/escravo, o roteador 7 é elegido porque o mestre devido a ele é um Roteador-ID mais alto. As bandeiras em etapas 13 e 14 mostram claramente que o roteador 7 é mestre (bandeira 0x7) e o roteador 6 é escravo (bandeira 0x2).

Na etapa 10, o roteador 6 recebe o pacote DBD e as transições da inicial do roteador 7 seu estado à 2-maneira.

Em etapa 12, o roteador 7 recebe o pacote DBD da inicial do roteador 6 e reconhece uma má combinação MTU. (O Roteador 7 é capaz de reconhecer uma incompatibilidade de MTU porque o Roteador 6 inclui a interface de MTU no campo de MTU da interface do pacote de DBD). A inicial DBD do roteador 6 é rejeitada pelo roteador que do roteador 7. 7 retransmite o pacote DBD inicial.

Etapas 13 mostra que o roteador 6, como o escravo, adota o número de sequência do roteador 7 e envia seu segundo pacote DBD que contém os encabeçamentos de seus LSA, que aumenta o tamanho do pacote. Contudo, o roteador 7 nunca recebe este pacote DBD porque é maior do que o roteador 7 MTU.

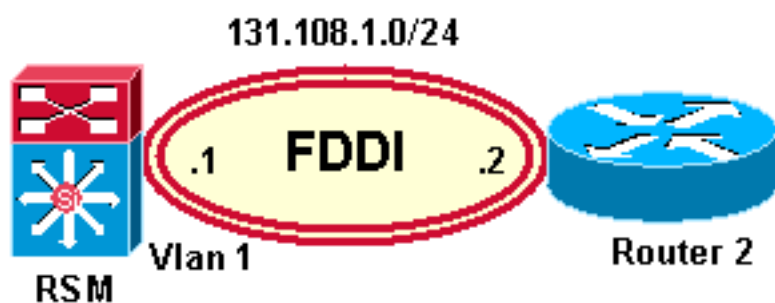
Após o passo 13, o roteador 7 continua retransmitindo o pacote DBD inicial ao roteador 6, enquanto o roteador 6 continua enviando os pacotes DBD que seguem o número de seqüência mestre. Este laço continua indefinidamente, que impede um ou outro roteador da transição fora do estado de intercâmbio/exstart.

A solução

Desde que o problema é causado pelo mtus incompatível, a solução é mudar um ou outro roteador MTU para combinar o vizinho MTU. Note que o Cisco IOS não apoia um chang do MTU físico em uma interface de LAN (tal como Ethernet ou Token Ring). Quando o problema ocorre entre um roteador Cisco e um outro roteador de fornecedor sobre media LAN, ajuste o MTU no roteador do outro fornecedor.

Nota: Detecção introduzida Cisco IOS Software Release 12.0(3) da má combinação da interface MTU. Esta detecção envolve o OSPF que anuncia a interface MTU nos pacotes DBD, que é de acordo com o [RFC 2178](#) OSPF , o apêndice G.9. [Quando um roteador recebe um pacote DBD anunciando um MTU maior do que pode receber, esse roteador ignora o pacote DBD e o estado do vizinho permanece em exstart. Isto impede a formação de uma adjacência. Para fixar este problema, assegure-se de que o MTU seja o mesmo no ambas as extremidades de um link.](#)

No Cisco IOS Software 12.01(3), o comando interface configuration do [ip ospf mtu-ignore](#) foi introduzido igualmente desligar a detecção de incompatibilidade do MTU; contudo, isto é precisado somente nas instâncias raras, segundo as indicações deste diagrama:



O diagrama acima mostra uma porta da Fiber Distributed Data Interface (FDDI) em um Cisco catalyst 5000 com um módulo de switch de rota (RS) conectado a uma interface do FDDI no roteador2. O LAN virtual (VLAN) no RS é uma interface de Ethernet virtual com um MTU de 1500, e a interface do FDDI no roteador2 tem um MTU de 4500. Quando um pacote é recebido na porta FDDI do interruptor, vai ao backplane e o FDDI à conversão/fragmentação dos Ethernet acontece dentro do interruptor próprio. Esta é uma instalação válida, mas com característica da detecção de incompatibilidade do MTU, a adjacência de OSPF não é formada entre o roteador e o RS. Como o FDDI e o Ethernet MTU são diferentes, este comando `ip ospf mtu-ignore` é útil na interface VLAN do RSM para que o OSPF pare de detectar incompatibilidade e de MTU e forme a adjacência.

É importante notar que a má combinação MTU, embora o mais comum, não é a única razão que os vizinhos de OSPF obtêm colados no estado de intercâmbio/exstart. O problema é causado com mais frequência pela incapacidade de trocar com êxito os pacotes DBD. Contudo, a causa de raiz podia ser algum do estes:

- Má combinação MTU
- O Unicast foi quebrado. No estado do exstart, o roteador envia um pacote do unicast ao vizinho para eleger o mestre e o escravo. Isso é verdade, exceto se você tem um link ponto a ponto; nesse caso, ele envia um pacote de transmissão múltipla. Estas são as possíveis causas: Mapeamento incorreto de VC (circuito virtual) em um ATM (Asynchronous Transfer Mode, Modo Assíncrono de Transferência) ou em um ambiente Frame Relay de rede altamente redundante. Problema de MTU, o que significa que os roteadores podem fazer ping apenas de pacotes com um determinado tamanho. A lista de acesso está bloqueando o pacote unicast. O NAT está sendo executado no roteador e está traduzindo o pacote do unicast.
- Vizinho entre o PRI e o BRI/dialer.
- Ambos os roteadores possuem o mesmo ID de roteador (erro de configuração).

Além, o [RFC 2328](#) OSPF, a seção 10.3, indica que o exstart/processo de intercâmbio está iniciado para alguns eventos (alguns de que poderia ser causado por problemas de software interno):

- SequenceNumberMismatch Número de sequência inesperado DD "Eu" bit sou ajustado inesperadamente Campo de opção diferente do campo de última opção recebido no pacote DBD
- BadLSReq O vizinho envia LSA não reconhecido durante um processo de troca. O vizinho pediu um LSA durante o processo de intercâmbio que não pode ser encontrado

Quando o OSPF não forma vizinhos, considere os fatores mencionados acima, como os meios físicos e o hardware de comunicação de rede, a fim de pesquisar defeitos o problema.

Informações Relacionadas

- [Estados vizinhos de OSPF](#)
- [Problemas vizinhos ao OSPF explicados](#)
- [Por que o comando show ip ospf neighbor revela vizinhos em estado init?](#)
- [Por que o Comando show ip ospf neighbor revela Vizinhos paralisados em Estado Bidirecional?](#)
- [Página de suporte de OSPF](#)
- [Suporte Técnico - Cisco Systems](#)