

# Pesquisa defeitos edições comuns EIGRP

## Índice

[Introdução](#)

[Informações de Apoio](#)

[Flapping vizinho](#)

[Problemas de rede](#)

[SIA](#)

[Expirado guardando o temporizador](#)

[Limite excedido da nova tentativa](#)

[Par reiniciado](#)

[Atualização inicial antes olá!](#)

[Outros problemas](#)

[Alterações de configuração](#)

[Autenticação](#)

[Má combinação em preliminar e em endereços IP secundários](#)

[DMVPN](#)

[Bandeiras explicadas](#)

[SIA](#)

[Definição do SIA](#)

[Sintomas](#)

[Possíveis causas](#)

[Dicas para Troubleshooting](#)

[Prefixos faltantes](#)

[Prefixos faltantes no RIB](#)

[Prefixo instalado pelo protocolo de roteamento com distância administrativa mais baixa](#)

[A Distribuir-lista obstrui o prefixo](#)

[Prefixos faltantes na tabela de topologia](#)

[Especificação da máscara para a saída apropriada do comando](#)

[O horizonte dividido obstrui o prefixo](#)

[Métrica](#)

[ID do roteador duplicado](#)

[Os valores K combinam mal/paradas oportunas](#)

[Balanceamento de carga dos custos desiguais \(variação\)](#)

[Vizinhos estáticos](#)

[Redistribuição da rota estática](#)

[Confiança e carga para o cálculo métrico](#)

[Alta utilização da CPU](#)

[EIGRP nas redes do Frame Relay \(fila de broadcast\)](#)

[Combinado mal COMO números](#)

[Resumo automático](#)

[Log de eventos EIGRP](#)

[A mesma rede aprendida por dois sistemas autônomo de EIGRP](#)

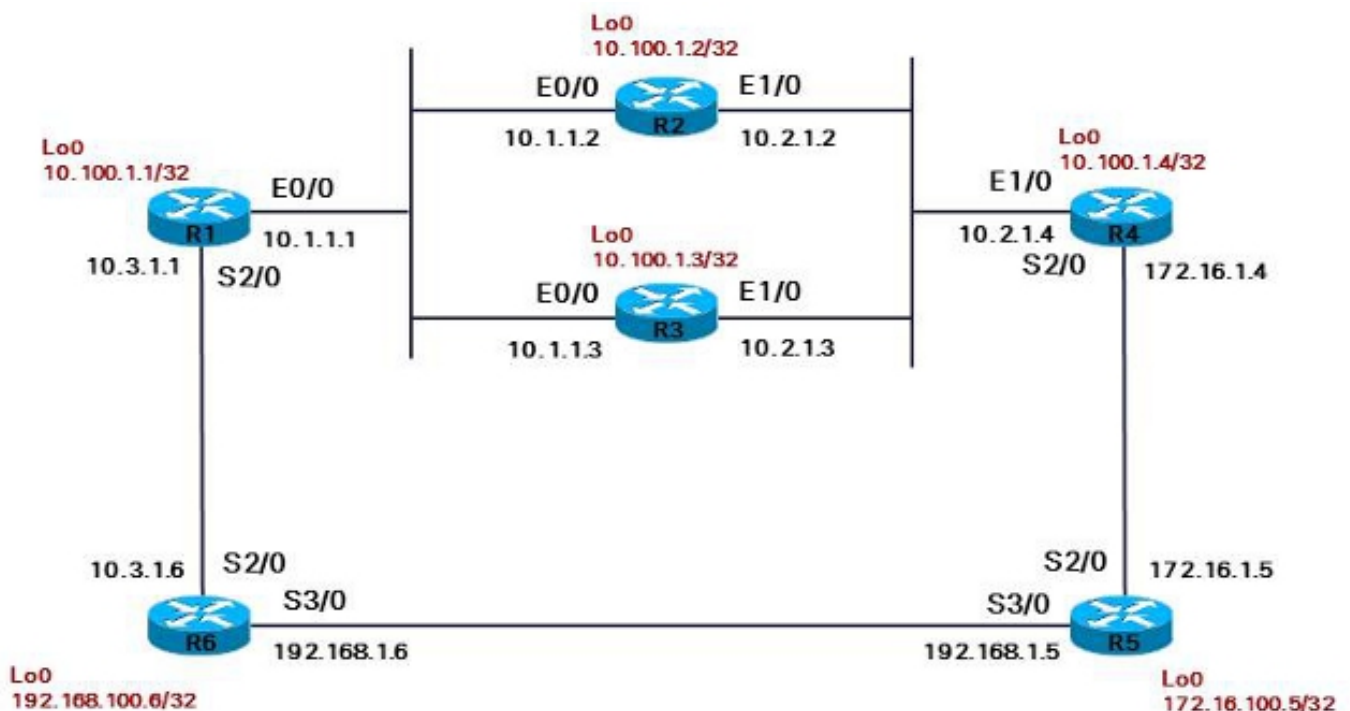
## Introdução

Este documento descreve como pesquisar defeitos as edições as mais comuns do Enhanced Interior Gateway Routing Protocol (EIGRP).

Nota: Este documento usa exemplos e aplicação no <sup>®</sup> do Cisco IOS a fim ilustrar os vários comportamentos que podem ser encontrados.

## Informações de Apoio

Esta é a topologia que é usada para este documento:



As seções que seguem descrevem algumas das edições as mais comuns EIGRP e de algumas pontas sobre como pesquisar defeitos as edições.

## Flapping vizinho

O único a maioria de problema comum que é encontrado com o uso do EIGRP é que não estabelece um neighborhood corretamente. Há diversas causas possíveis para esta:

- Edição da unidade de transmissão máxima (MTU)

- Uma comunicação de sentido único (enlaces unidirecional)
- Há um problema de transmissão múltipla no link
- Problemas do unicast
- Problemas de qualidade do link
- Edições da autenticação
- Edições do Misconfiguration

Se você não recebe uma mensagem dos hello de EIGRP, você não pode ver o vizinho na lista vizinha. Inscreva o **comando show ip eigrp neighbors** a fim ver a informação do vizinho EIGRP e identificar a edição:

```
R2#show ip eigrp neighbors
```

```
IP-EIGRP neighbors for process 1
H   Address   Interface Hold Uptime   SRTT   RTO   Q   Seq
      (sec)      (ms)      Cnt Num
3   10.1.1.1   Et0/0     12 00:00:48   1   5000   1   0
2   10.1.1.3   Et0/0     12 02:47:13   22   200   0   339
1   10.2.1.4   Et1/0     12 02:47:13   24   200   0   318
0   10.2.1.3   Et1/0     12 02:47:13   20   200   0   338 13   20   200   0   338
```

Se você pensa que o neighborhood esteve formado, mas você não tem os prefixos que você deve aprender desse vizinho, verifique a saída do comando precedente: Se a *Q-contagem* é sempre diferente de zero, poderia ser uma indicação que os mesmos pacotes EIGRP estão retransmitidos continuamente. Inscreva o **comando detail dos vizinhos EIGRP da mostra IP** a fim verificar se o mesmo pacote está enviado sempre. Se o número de sequência do primeiro pacote é sempre o mesmo, a seguir o mesmo pacote está retransmitido indefinidamente:

```
R2#show ip eigrp neighbors detail
```

```
IP-EIGRP neighbors for process 1
H   Address           Interface           Hold Uptime   SRTT   RTO   Q   Seq
      (sec)              (ms)              Cnt Num
3   10.1.1.1           Et0/0              11 00:00:08   1   4500   1   0
  Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack
  UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced
2   10.1.1.3           Et0/0              11 02:47:56   22   200   0   339
  Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1   10.2.1.4           Et1/0              10 02:47:56   24   200   0   318
  Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0   10.2.1.3           Et1/0              11 02:47:56   20   200   0   338
  Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2
```

Você pode ver na saída que o primeiro vizinho tem um problema, e o *Uptime* está restaurado.

É importante que você verifica se o roteador EIGRP do processo tem o **comando eigrp log-neighbor-changes**. Contudo, isto é incluído à revelia desde a identificação de bug Cisco [CSCdx67706](#), assim que não aparece na configuração nesse caso. Verifique a entrada nos logs para ver se há ambos os vizinhos EIGRP em cada lado do link. Pelo menos em um dos logs, deve haver uma entrada significativa.

Estão aqui todas as razões possíveis para uma mudança da vizinhança do EIGRP e suas entradas de registro:

- Nenhum pacote EIGRP foi recebido durante o tempo de contenção:

R2#show ip eigrp neighbors detail

```
IP-EIGRP neighbors for process 1
H   Address                Interface          Hold Uptime      SRTT   RTO  Q  Seq
                               (sec)            (ms)            Cnt Num
3   10.1.1.1                Et0/0             11 00:00:08     1  4500  1  0
    Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack
    UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced
2   10.1.1.3                Et0/0             11 02:47:56     22  200  0  339
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1   10.2.1.4                Et1/0             10 02:47:56     24  200  0  318
    Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0   10.2.1.3                Et1/0             11 02:47:56     20  200  0  338
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2
```

- Um pacote seguro EIGRP não foi reconhecido dentro do limite da nova tentativa:

R2#show ip eigrp neighbors detail

```
IP-EIGRP neighbors for process 1
H   Address                Interface          Hold Uptime      SRTT   RTO  Q  Seq
                               (sec)            (ms)            Cnt Num
3   10.1.1.1                Et0/0             11 00:00:08     1  4500  1  0
    Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack
    UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced
2   10.1.1.3                Et0/0             11 02:47:56     22  200  0  339
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1   10.2.1.4                Et1/0             10 02:47:56     24  200  0  318
    Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0   10.2.1.3                Et1/0             11 02:47:56     20  200  0  338
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2
```

- O EIGRP vê a relação em um *estado inativo*:

R2#show ip eigrp neighbors detail

```
IP-EIGRP neighbors for process 1
H   Address                Interface          Hold Uptime      SRTT   RTO  Q  Seq
                               (sec)            (ms)            Cnt Num
3   10.1.1.1                Et0/0             11 00:00:08     1  4500  1  0
    Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack
    UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced
2   10.1.1.3                Et0/0             11 02:47:56     22  200  0  339
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1   10.2.1.4                Et1/0             10 02:47:56     24  200  0  318
    Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0   10.2.1.3                Et1/0             11 02:47:56     20  200  0  338
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2
```

- O roteador recebeu um pacote de atualização inicial e reiniciou o neighborship:

R2#show ip eigrp neighbors detail

```
IP-EIGRP neighbors for process 1
H   Address                Interface          Hold Uptime      SRTT   RTO  Q  Seq
                               (sec)            (ms)            Cnt Num
3   10.1.1.1                Et0/0             11 00:00:08     1  4500  1  0
    Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack
    UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced
2   10.1.1.3                Et0/0             11 02:47:56     22  200  0  339
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1   10.2.1.4                Et1/0             10 02:47:56     24  200  0  318
    Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0   10.2.1.3                Et1/0             11 02:47:56     20  200  0  338
    Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2
```

- O roteador recebeu um pacote de atualização inicial e formou uma adjacência nova:

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- O comando `ip eigrp neighbor claro` foi inscrito, que conduziu a um *claro manual*:

R2#`show ip eigrp neighbors detail`

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- O endereço IP de Um ou Mais Servidores Cisco ICM NT na relação foi mudado:

R2#`show ip eigrp neighbors detail`

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- Havia um atraso/alteração de largura de banda na relação: Nota: Isto ocorre somente em umas versões de código mais velhas. Não há nenhum flap vizinho desde a identificação de bug Cisco [CSCdp08764](#). R2#`show ip eigrp neighbors detail`

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- Os valores K são desconfigurados ou uma *parada oportuna* ocorre:

R2#`show ip eigrp neighbors detail`

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- Uma parada oportuna ocorre:

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- O comando md5 do eigrp 1 do modo de autenticação IP foi configurado na relação:

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- Um reinício gracioso/transmissão sem parar (NSF) ocorreu:

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H   | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |
|---|----------|-----------|----------------------|--------------|------|----------|------------|
| 3   | 10.1.1.1 | Et0/0     | 11 00:00:08          | 1            | 4500 | 1        | 0          |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |           |                      |              |      |          |            |
| 2   | 10.1.1.3 | Et0/0     | 11 02:47:56          | 22           | 200  | 0        | 339        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |           |                      |              |      |          |            |
| 1   | 10.2.1.4 | Et1/0     | 10 02:47:56          | 24           | 200  | 0        | 318        |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |           |                      |              |      |          |            |
| 0   | 10.2.1.3 | Et1/0     | 11 02:47:56          | 20           | 200  | 0        | 338        |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |           |                      |              |      |          |            |

- Os vizinhos a que há umas perguntas enviadas sem uma resposta recebida são cancelados:

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H | Address | Interface | Hold Uptime | SRTT | RTO | Q | Seq |
|---|---------|-----------|-------------|------|-----|---|-----|
|---|---------|-----------|-------------|------|-----|---|-----|

|   |          |       | (sec)       | (ms) | Cnt  | Num   |
|---|----------|-------|-------------|------|------|-------|
| 3   | 10.1.1.1 | Et0/0 | 11 00:00:08 | 1    | 4500 | 1 0   |
| Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced |          |       |             |      |      |       |
| 2   | 10.1.1.3 | Et0/0 | 11 02:47:56 | 22   | 200  | 0 339 |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10   |          |       |             |      |      |       |
| 1   | 10.2.1.4 | Et1/0 | 10 02:47:56 | 24   | 200  | 0 318 |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8  |          |       |             |      |      |       |
| 0   | 10.2.1.3 | Et1/0 | 11 02:47:56 | 20   | 200  | 0 338 |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2  |          |       |             |      |      |       |

## Problemas de rede

Estas cinco edições indicam um problema de rede:

- Um estado (SIA) Colar-Em-ativo
- Um temporizador guardando expirado
- Um limite excedido da nova tentativa
- Um par reiniciado
- Uma atualização inicial é enviada antes do pacote Hello

## SIA

Refira a seção [SIA](#) deste documento.

## Expirado guardando o temporizador

Um temporizador guardando expirado indica que o roteador não recebeu nenhum pacote EIGRP (isto é, um hello de EIGRP ou algum outro pacote EIGRP) durante o intervalo do hold-time. Há mais do que provavelmente um problema no link neste caso.

Certifique-se do roteador receba os pacotes de hello de EIGRP neste link e que o outro lado os envia. A fim verificar isto, incorpore o comando do **pacote EIGRP debug olá!**

Como uma alternativa ao uso do comando debug, você pode sibilhar o endereço IP 224.0.0.10 e verificar se esse vizinho responde.

As causas possíveis para o problema de transmissão múltipla no link são devidas conectar problemas, como se um interruptor intermediário obstrui os pacotes de hello de EIGRP.

Um outro teste rápido que você possa executar é tentar um outro protocolo que use um outro endereço IP multicast. Por exemplo, você pode configurar a versão 2 do Routing Information Protocol (RIP) que usa o endereço IP multicast 224.0.0.9.

## Limite excedido da nova tentativa

Um limite excedido da nova tentativa indica que um pacote seguro EIGRP não esteve reconhecido épocas múltiplas. Um pacote seguro EIGRP é um destes cinco tipos de pacotes:

- Atualizar
- Consulta
- Resposta
- SIA-pergunta
- SIA-resposta

O pacote EIGRP seguro foi retransmitido pelo menos 16 vezes. Um pacote é cada retransmitido retransmite o Time Out (RTO). O mínimo RTO é a Senhora 200 e o máximo é a Senhora 5,000. O RTO aumenta ou diminui dinamicamente através da observação da diferença de horário entre o tempo que o pacote EIGRP seguro está enviado e o tempo que o reconhecimento está recebido. Quando o pacote seguro não é reconhecido, o RTO aumenta. Se isto persiste, a seguir o RTO aumenta até cinco segundos rapidamente, assim que o limite da nova tentativa pode alcançar 16 segundos x 5 = 80 segundos. Contudo, se o tempo de contenção EIGRP é maior de 80 segundos, o neighborhood não vai para baixo até que o tempo de contenção expire. Isto pode ocorrer nos enlaces de WAN lentos onde, por exemplo, o tempo de contenção do padrão é 180 segundos.

Para os links com o tempo de contenção abaixo do que 80 segundos, isto significa eficazmente que se o tempo de contenção não expira, está prosseguido pelos pacotes de hello de EIGRP. O limite da nova tentativa pode então ser excedido. Isto indica que há um problema com MTU ou um problema do unicast. Os pacotes de hello de EIGRP são pequenos; (o primeiro) pacote da atualização EIGRP pode ser até o MTU completo. Será tamanho do MTU completo se há bastante prefixos para encher a atualização. O vizinho pode ser instruído através da recepção dos pacotes de hello de EIGRP, mas a adjacência total não pôde suceder se o pacote da atualização EIGRP não é reconhecido.

Tipicamente, esta é a saída que aparece:

```
R2#show ip eigrp neighbors detail
```

```
IP-EIGRP neighbors for process 1
H   Address                Interface      Hold Uptime    SRTT   RTO  Q  Seq
                               (sec)         (ms)          Cnt Num
3   10.1.1.1                Et0/0         11 00:00:08    1  4500  1  0
  Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack
  UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced
2   10.1.1.3                Et0/0         11 02:47:56    22   200  0  339
  Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1   10.2.1.4                Et1/0         10 02:47:56    24   200  0  318
  Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0   10.2.1.3                Et1/0         11 02:47:56    20   200  0  338
  Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2
```

Nota: Até à data da identificação de bug Cisco [CSCsc72090](#), o EIGRP igualmente usa as *configurações MTU IP da relação*. Antes que este reparo esteve aplicado, os pacotes EIGRP tornaram-se fragmentados se o IP MTU foi configurado com um valor que fosse mais baixo de 1,500. Esta edição pode tipicamente ocorrer em redes do Dynamic Multipoint VPN (DMVPN).



Uma segunda possibilidade é que os pacotes de hello de EIGRP a fazem porque multicasted ao endereço IP 224.0.0.10. Alguns pacotes da atualização EIGRP puderam fazê-la, como podem ser multicasted. Contudo, os pacotes seguros retransmitidos EIGRP são sempre unicast. Se o trajeto de dados do unicast ao vizinho é quebrado, o pacote seguro retransmitido não processa corretamente. Sibile o endereço IP unicast do vizinho EIGRP (com o tamanho do sibilo ajuste ao máximo o tamanho do MTU do link, e com não fragmentam o bit (DF-bit) se ajustam) a fim verificar.

Um link de sentido único pode causar este problema também. O EIGRP Router pôde receber os pacotes de hello de EIGRP, mas os pacotes que são enviados deste vizinho não os fazem através do link. Se os pacotes Hello não o fazem, o roteador é inconsciente porque os pacotes Hello são enviados incerta. Os pacotes da atualização EIGRP que são enviados não serão reconhecidos.

Os pacotes seguros EIGRP ou o reconhecimento podem tornar-se corrompidos. Um teste rápido é enviar sibilos com a validação da resposta permitida:

```
R1#ping
Protocol [ip]:
Target IP address: 10.1.1.2
Repeat count [5]: 10
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]: yes
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
Reply data will be validated
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 1/24/152 ms
```

**Permita o comando `debug eigrp packets` a fim verificar a transmissão e a recepção dos pacotes de hello de EIGRP e dos pacotes da atualização EIGRP em um mínimo:**

```
R1#debug eigrp packets ?

SIAquery  EIGRP SIA-Query packets
SIAreply  EIGRP SIA-Reply packets
ack       EIGRP ack packets
hello     EIGRP hello packets
ipxsap    EIGRP ipxsap packets
probe     EIGRP probe packets
query     EIGRP query packets
reply     EIGRP reply packets
request   EIGRP request packets
retry     EIGRP retransmissions
stub      EIGRP stub packets
terse     Display all EIGRP packets except Hellos
update    EIGRP update packets
verbose   Display all EIGRP packets
```

Está aqui um exemplo típico da edição *excedida limite da nova tentativa*:

```
R2#show ip eigrp neighbors
```

IP-EIGRP neighbors for process 1

| H | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |    |    |     |       |
|---|----------|-----------|----------------------|--------------|------|----------|------------|----|----|-----|-------|
| 3 | 10.1.1.1 | Et0/0     | 12 00:00:48          | 1            | 5000 | 1        | 0          |    |    |     |       |
| 2 | 10.1.1.3 | Et0/0     | 12 02:47:13          | 22           | 200  | 0        | 339        |    |    |     |       |
| 1 | 10.2.1.4 | Et1/0     | 12 02:47:13          | 24           | 200  | 0        | 318        |    |    |     |       |
| 0 | 10.2.1.3 | Et1/0     | 12 02:47:13          | 20           | 200  | 0        | 338        | 13 | 20 | 200 | 0 338 |

Nota: Há sempre uns ou vários pacotes na fila (Q Cnt).

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H  | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |  |  |  |  |
|--|----------|-----------|----------------------|--------------|------|----------|------------|--|--|--|--|
| 3  | 10.1.1.1 | Et0/0     | 10 00:00:59          | 1            | 5000 | 1        | 0          |  |  |  |  |
| Version 12.4/1.2, Retrans: 12, Retries: 12, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 349 ser 0-0 Sent 59472 Init Sequenced |          |           |                      |              |      |          |            |  |  |  |  |
| 2  | 10.1.1.3 | Et0/0     | 11 02:47:23          | 22           | 200  | 0        | 339        |  |  |  |  |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10  |          |           |                      |              |      |          |            |  |  |  |  |
| 1  | 10.2.1.4 | Et1/0     | 11 02:47:23          | 24           | 200  | 0        | 318        |  |  |  |  |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8   |          |           |                      |              |      |          |            |  |  |  |  |
| 0  | 10.2.1.3 | Et1/0     | 10 02:47:23          | 20           | 200  | 0        | 338        |  |  |  |  |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2   |          |           |                      |              |      |          |            |  |  |  |  |

Segundo as indicações da saída, o R2 espera o primeiro pacote de atualização (jogo do bit do init) do vizinho no endereço IP 10.1.1.1.

Nesta saída seguinte, o R2 espera o reconhecimento do primeiro pacote de atualização (jogo do bit do init) do vizinho no endereço IP 10.1.1.1.

Nota: O RTO está em seu máximo da Senhora 5,000, que indica que os pacotes seguros EIGRP não estão reconhecidos dentro dos cinco segundos.

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H  | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |  |  |  |  |
|--|----------|-----------|----------------------|--------------|------|----------|------------|--|--|--|--|
| 3  | 10.1.1.1 | Et0/0     | 11 00:01:17          | 1            | 5000 | 1        | 0          |  |  |  |  |
| Version 12.4/1.2, Retrans: 16, Retries: 16, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 349 ser 0-0 Sent 77844 Init Sequenced |          |           |                      |              |      |          |            |  |  |  |  |
| 2  | 10.1.1.3 | Et0/0     | 12 02:47:42          | 22           | 200  | 0        | 339        |  |  |  |  |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10  |          |           |                      |              |      |          |            |  |  |  |  |
| 1  | 10.2.1.4 | Et1/0     | 10 02:47:42          | 24           | 200  | 0        | 318        |  |  |  |  |
| Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8   |          |           |                      |              |      |          |            |  |  |  |  |
| 0  | 10.2.1.3 | Et1/0     | 11 02:47:42          | 20           | 200  | 0        | 338        |  |  |  |  |
| Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2   |          |           |                      |              |      |          |            |  |  |  |  |

O número de retransmissões aumenta firmemente. É sempre o mesmo pacote na fila (349 segs.s). Depois que o R2 enviou a isto o mesmo pacote 16 vezes, o neighborhood vai para baixo:

R2#show ip eigrp neighbors detail

IP-EIGRP neighbors for process 1

| H  | Address  | Interface | Hold Uptime<br>(sec) | SRTT<br>(ms) | RTO  | Q<br>Cnt | Seq<br>Num |  |  |  |  |
|--|----------|-----------|----------------------|--------------|------|----------|------------|--|--|--|--|
| 3  | 10.1.1.1 | Et0/0     | 11 00:01:17          | 1            | 5000 | 1        | 0          |  |  |  |  |
| Version 12.4/1.2, Retrans: 16, Retries: 16, Waiting for Init, Waiting for Init Ack<br>UPDATE seq 349 ser 0-0 Sent 77844 Init Sequenced |          |           |                      |              |      |          |            |  |  |  |  |
| 2  | 10.1.1.3 | Et0/0     | 12 02:47:42          | 22           | 200  | 0        | 339        |  |  |  |  |

```

Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1 10.2.1.4 Et1/0 10 02:47:42 24 200 0 318
Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0 10.2.1.3 Et1/0 11 02:47:42 20 200 0 338
Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2

```

O processo começa mais uma vez:

```
R2#show ip eigrp neighbors detail
```

```

IP-EIGRP neighbors for process 1
H Address Interface Hold Uptime SRTT RTO Q Seq
(sec) (ms) Cnt Num
3 10.1.1.1 Et0/0 11 00:00:08 1 4500 1 0
Version 12.4/1.2, Retrans: 2, Retries: 2, Waiting for Init, Waiting for Init Ack
UPDATE seq 350 ser 0-0 Sent 8040 Init Sequenced
2 10.1.1.3 Et0/0 11 02:47:56 22 200 0 339
Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 10
1 10.2.1.4 Et1/0 10 02:47:56 24 200 0 318
Version 12.4/1.2, Retrans: 10, Retries: 0, Prefixes: 8
0 10.2.1.3 Et1/0 11 02:47:56 20 200 0 338
Version 12.4/1.2, Retrans: 11, Retries: 0, Prefixes: 2

```

A saída do comando **sóbrio dos pacotes EIGRP debugar** mostra que o R2 envia o mesmo pacote a toda hora:

Nota: Os aumentos do valor de nova tentativa, o valor das bandeiras são **0x1**, e o *bit de Init* é ajustado.

```
R2#debug eigrp packets terse
```

```

EIGRP Packets debugging is on
(UPDATE, REQUEST, QUERY, REPLY, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)

```

```

R2#
EIGRP: Sending UPDATE on Ethernet0/0 nbr 10.1.1.1, retry 14, RTO 5000
AS 1, Flags 0x1, Seq 350/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/1
EIGRP: Sending UPDATE on Ethernet0/0 nbr 10.1.1.1, retry 15, RTO 5000
AS 1, Flags 0x1, Seq 350/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/1

```

O tempo de contenção não expira porque os pacotes Hello são enviados e recebidos corretamente:

```

R2#debug eigrp packets hello
EIGRP Packets debugging is on
(HELLO)

```

```

EIGRP: Received HELLO on Ethernet0/0 nbr 10.1.1.1
AS 1, Flags 0x0, Seq 0/0 idbQ 0/0

```

## Par reiniciado

Se você observa um **par reiniciado** repetidamente em um roteador, indica que o roteador recebe os pacotes de atualização iniciais de seu vizinho. Esteja ciente da **bandeira 1** nos pacotes de atualização recebidos.

```
R2#deb eigrp packets terse
```

```

EIGRP Packets debugging is on
(UPDATE, REQUEST, QUERY, REPLY, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)

```

```
R2#
EIGRP: Received Sequence TLV from 10.1.1.1
    10.1.1.2
    address matched
    clearing CR-mode
EIGRP: Received CR sequence TLV from 10.1.1.1, sequence 479
EIGRP: Received UPDATE on Ethernet0/0 nbr 10.1.1.1
    AS 1, Flags 0xA, Seq 479/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0,
not in CR-mode, packet discarded
EIGRP: Received UPDATE on Ethernet0/0 nbr 10.1.1.1
    AS 1, Flags 0x1, Seq 478/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0

%DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1 (Ethernet0/0) is down:
peer restarted
%DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1 (Ethernet0/0) is up:
new adjacency
EIGRP: Enqueueing UPDATE on Ethernet0/0 nbr 10.1.1.1 iidbQ un/rely 0/1
peerQ un/rely 0/0
```

## Rubrique a atualização antes olá!

Está aqui um exemplo onde o pacote de atualização inicial seja recebido antes do pacote Hello:

```
R2#deb eigrp packets terse

EIGRP Packets debugging is on
    (UPDATE, REQUEST, QUERY, REPLY, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)

R2#
EIGRP: Received Sequence TLV from 10.1.1.1
    10.1.1.2
    address matched
    clearing CR-mode
EIGRP: Received CR sequence TLV from 10.1.1.1, sequence 479
EIGRP: Received UPDATE on Ethernet0/0 nbr 10.1.1.1
    AS 1, Flags 0xA, Seq 479/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0,
not in CR-mode, packet discarded
EIGRP: Received UPDATE on Ethernet0/0 nbr 10.1.1.1
    AS 1, Flags 0x1, Seq 478/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0

%DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1 (Ethernet0/0) is down:
peer restarted
%DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1 (Ethernet0/0) is up:
new adjacency
EIGRP: Enqueueing UPDATE on Ethernet0/0 nbr 10.1.1.1 iidbQ un/rely 0/1
peerQ un/rely 0/0
```

Se isto ocorre uma vez depois que um flap vizinho, a seguir esta situação não é um problema. Contudo, se você a experimenta frequentemente, indica que o unicast no link é operacional, mas o Multicast no link é quebrado. Ou seja o roteador recebe o pacote de atualização do unicast, mas não os pacotes Hello.

## Outros problemas

Alguns outros tipos de problema incluem:

- Alterações de configuração
- Edições da autenticação

- Más combinações em preliminar e em endereços IP secundários
- Edições DMVPN

Estas edições são explicadas em maiores detalhes nas seções que seguem.

## Alterações de configuração

Nota: Os resultados dos comandos que são usados durante todo esta seção são os mesmos se você configura a negação pelo contrário (*nenhum* comando).

Quando você configura a indicação sumária (ou o *resumo automático*) na relação, você observa esta mensagem no roteador:

```
R2#deb eigrp packets terse

EIGRP Packets debugging is on
  (UPDATE, REQUEST, QUERY, REPLY, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)

R2#
EIGRP: Received Sequence TLV from 10.1.1.1
      10.1.1.2
      address matched
      clearing CR-mode
EIGRP: Received CR sequence TLV from 10.1.1.1, sequence 479
EIGRP: Received UPDATE on Ethernet0/0 nbr 10.1.1.1
      AS 1, Flags 0xA, Seq 479/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0,
not in CR-mode, packet discarded
EIGRP: Received UPDATE on Ethernet0/0 nbr 10.1.1.1
      AS 1, Flags 0x1, Seq 478/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0

%DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1 (Ethernet0/0) is down:
peer restarted
%DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1 (Ethernet0/0) is up:
new adjacency
EIGRP: Enqueueing UPDATE on Ethernet0/0 nbr 10.1.1.1 iidbQ un/rely 0/1
peerQ un/rely 0/0
```

Está aqui um exemplo que mostre a configuração de uma *lista de distribuição global* para o processo de EIGRP:

```
R1(config-router)#distribute-list 1 out
R1(config-router)#
```

Esta mensagem é observada no roteador:

Nota: O mesmo ocorre quando você configura um *<> da distribuir-lista dentro* também.

```
R1(config-router)#distribute-list 1 out
R1(config-router)#
```

Todos os vizinhos EIGRP vão então abaixo de quando você configura uma distribuir-lista da *relação* para o processo de EIGRP:

```
R1(config-router)#distribute-list 1 out
R1(config-router)#
```

Neste caso, somente as vizinhanças do EIGRP nesta relação são restauradas.

Nota: Após a identificação de bug Cisco [CSCdy20284](#), os neighborships não são restaurados para alterações manual tais como a sumarização e os filtros.

## Autenticação

A autenticação pode ser desconfigurada ou faltar. Isto pode fazer com que a vizinhança do EIGRP vá para baixo devido ao novo-limite excedido. Permita o **comando debug eigrp packets** a fim confirmar que é a autenticação do message digest 5 (MD5) que causa a edição:

```
R1#debug eigrp packets
```

```
EIGRP Packets debugging is on
  (UPDATE, REQUEST, QUERY, REPLY, HELLO, IPXSAP, PROBE, ACK, STUB, SIAQUERY,
  SIAREPLY)
```

```
EIGRP: Ethernet0/0: ignored packet from 10.1.1.3, opcode = 1 (missing
authentication or key-chain missing)
```

## Má combinação em preliminar e em endereços IP secundários

O EIGRP manda olá! e todos pacotes restantes do endereço IP primário. Os pacotes estão aceitados do outro roteador se os endereços IP de origem caem na escala de endereço IP primário ou em essa das escalas de endereço IP secundário na relação. Se não, este Mensagem de Erro (quando os log-vizinho-avisos do eigrp são permitidos) é observado:

```
R1#debug eigrp packets
```

```
EIGRP Packets debugging is on
  (UPDATE, REQUEST, QUERY, REPLY, HELLO, IPXSAP, PROBE, ACK, STUB, SIAQUERY,
  SIAREPLY)
```

```
EIGRP: Ethernet0/0: ignored packet from 10.1.1.3, opcode = 1 (missing
authentication or key-chain missing)
```

## DMVPN

Verifique para ver se há problemas de IPsec nas redes de DMVPN. O IPsec pode fazer com que o EIGRP bata se a criptografia não está limpa:

```
show crypto ipsec sa
```

```
protected vrf:
local  ident (addr/mask/prot/port):  (10.10.110.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port):  (10.10.101.1/255.255.255.255/47/0)
current_peer: 144.23.252.1:500
  PERMIT, flags={origin_is_acl,}
#pkts encaps: 190840467, #pkts encrypt: 190840467, #pkts digest 190840467
#pkts decaps: 158102457, #pkts decrypt: 158102457, #pkts verify 158102457
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 5523, #recv errors 42
```

## Bandeiras explicadas

Há um campo de 32 bits das *bandeiras no* encabeçamento de pacote EIGRP, e é útil compreender as indicações dos vários valores da bandeira.

- **Bit da bandeira 0x1 Init**

Esta bandeira é ajustada no pacote de atualização inicial.

```
show crypto ipsec sa
```

```
protected vrf:
local  ident (addr/mask/prot/port):  (10.10.110.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port):  (10.10.101.1/255.255.255.255/47/0)
current_peer: 144.23.252.1:500
  PERMIT, flags={origin_is_acl,}
#pkts encaps: 190840467, #pkts encrypt: 190840467, #pkts digest 190840467
#pkts decaps: 158102457, #pkts decrypt: 158102457, #pkts verify 158102457
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 5523, #recv errors 42
```

- **Bandeira 0x2**

Esta bandeira indica que condicional receba o modo (CR-MODE). Isto é parte do processo seguro do Multicast EIGRP e é usado a fim permitir os vizinhos que não reconheceram um pacote seguro precedente para alcançar em um link compartilhado. Os endereços no Type Length Value da sequência (TLV) são os pares que devem ignorar os pacotes de transmissão múltipla até que alcancem através dos pacotes do unicast.

```
show crypto ipsec sa
```

```
protected vrf:
local  ident (addr/mask/prot/port):  (10.10.110.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port):  (10.10.101.1/255.255.255.255/47/0)
current_peer: 144.23.252.1:500
  PERMIT, flags={origin_is_acl,}
#pkts encaps: 190840467, #pkts encrypt: 190840467, #pkts digest 190840467
#pkts decaps: 158102457, #pkts decrypt: 158102457, #pkts verify 158102457
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 5523, #recv errors 42
```

- **Bandeira 0x4**

Esta bandeira é o bit do reinício (RS mordido). Está ajustada nos pacotes Hello e nos pacotes de atualização quando o NSF é sinalizado. Um roteador NSF-ciente vê este bit a fim detectar se o roteador vizinho reinicia. O vizinho que detecta então sabe para prosseguir a adjacência EIGRP. O roteador que reinicia vistas esta bandeira a fim determinar se o par ajuda com o reinício.

```
show crypto ipsec sa
```

```
protected vrf:
local  ident (addr/mask/prot/port):  (10.10.110.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port):  (10.10.101.1/255.255.255.255/47/0)
current_peer: 144.23.252.1:500
  PERMIT, flags={origin_is_acl,}
#pkts encaps: 190840467, #pkts encrypt: 190840467, #pkts digest 190840467
#pkts decaps: 158102457, #pkts decrypt: 158102457, #pkts verify 158102457
```

```
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 5523, #recv errors 42
```

- **Bandeira 0x8**

Este é o bit da Fim--tabela (EOT). Este bit indica que a tabela de roteamento completa esteve enviada ao vizinho. Um roteador NSF-capaz vê este bit a fim determinar se o roteador vizinho terminou seu reinício. Um roteador NSF-capaz espera este bit antes que remova as rotas velhas do roteador que reinicia.

`show crypto ipsec sa`

```
protected vrf:
local ident (addr/mask/prot/port): (10.10.110.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port): (10.10.101.1/255.255.255.255/47/0)
current_peer: 144.23.252.1:500
  PERMIT, flags={origin_is_acl,}
#pkts encaps: 190840467, #pkts encrypt: 190840467, #pkts digest 190840467
#pkts decaps: 158102457, #pkts decrypt: 158102457, #pkts verify 158102457
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 5523, #recv errors 42
```

As bandeiras são imprimidas em uma ENCANTAM o número. Assim, a bandeira 0x5 significa que as bandeiras 4 e 1 estão ajustadas; A bandeira 0x9 significa que as bandeiras 8 e 1 estão ajustadas; A bandeira 0xA significa que as bandeiras 8 e 2 estão ajustadas.

Você pode usar estes comandos a fim pesquisar defeitos vizinhos do flapping:

- **mostre o detalhe da relação do eigrp**
- **mostre o detalhe do vizinho EIGRP IP**
- **sibile o unicast**
- **sibile com tamanho MTU completo**
- **o sibilo com “verifica dados da resposta”**
- **Multicast do sibilo**
- **debugar o pacote EIGRP (olá!)**
- **mostre o tráfego do eigrp IP**
- **mostre o tráfego IP | comece o EIGRP**

## SIA

Esta seção fornece uma vista geral do estado SIA, de alguns sintomas possíveis e de causas, e como pesquisá-la defeitos.



## Definição do SIA

O estado SIA significa que um EIGRP Router não recebeu uma resposta a uma pergunta de uns ou vários vizinhos dentro do tempo distribuído (aproximadamente três minutos). Quando isto ocorre, o EIGRP cancela os vizinhos que não enviam uma resposta e registra um Mensagem de Erro **DUAL-3-SIA** para a rota que foi active.

## Sintomas

Estas mensagens podem ser consideradas em um ou muito Roteadores:

```
show crypto ipsec sa
```

```
protected vrf:
local ident (addr/mask/prot/port): (10.10.110.1/255.255.255.255/47/0)
remote ident (addr/mask/prot/port): (10.10.101.1/255.255.255.255/47/0)
current_peer: 144.23.252.1:500
  PERMIT, flags={origin_is_acl,}
#pkts encaps: 190840467, #pkts encrypt: 190840467, #pkts digest 190840467
#pkts decaps: 158102457, #pkts decrypt: 158102457, #pkts verify 158102457
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 5523, #recv errors 42
```

Se isto ocorre somente esporadicamente, pode ser ignorado. Se ocorre frequentemente, indica um problema de rede persistente.

## Possíveis causas

Estão aqui algumas causas possíveis para um estado SIA:

- Links não sincronizados
- Links ruins
- Rotas não sincronizadas
- Links congestionados
- Diâmetro da rede grande (grande escala da pergunta)
- Falta de memória
- Alta utilização da CPU
- Misconfiguration (valor de largura de banda errado)

## [Dicas para Troubleshooting](#)

Quando uma situação SIA ocorre, há um problema em algum lugar na rede. A causa exata pode

ser difícil de descobrir. Há duas aproximações:

- Veja os prefixos que são relatados consistentemente como o SIA e determine as normalizações.
- Encontre o roteador que consistentemente não responde a perguntas para estas rotas.

Determine se todos os prefixos para que o SIA é relatado têm normalizações. Por exemplo, todos os puderam ser rotas de /32 da borda da rede (como em redes dial-up). Em caso afirmativo, pôde indicar o lugar do problema na rede (a saber, onde estes prefixos originados).

Finalmente, você deve descobrir o lugar onde os ou mais roteadores enviam perguntas e não recebem respostas, quando o roteador downstream não estiver neste estado. Por exemplo, o roteador poderia enviar perguntas e são reconhecidos, mas a resposta do roteador downstream não é recebida.

Você pode usar o **comando show ip eigrp topology active** a fim ajudar a pesquisar defeitos a edição SIA. Procure o r pequeno na saída do comando. Isto significa que o roteador espera uma resposta a uma pergunta para esse prefixo desse vizinho.

Exemplo: Veja a topologia. Os links R1-R6 e R1-R5 são fechados. Quando a interface de loopback do r1 do roteador é fechada, o r1 envia uma pergunta para o prefixo 10.100.1.1/32 ao R2 e ao R3. O r1 do roteador é agora ativo para este prefixo. O Roteadores R2 e R3 vai active e pergunta por sua vez o roteador R4, que vai active e envia uma pergunta ao R5. O roteador R5 vai finalmente active e envia uma pergunta ao R6. O roteador R6 deve retornar uma resposta ao R5. O roteador R5 vai voz passiva e responde ao R4, que vai por sua vez voz passiva e envia uma resposta ao R2 e ao R3. Finalmente, o R2 e o R3 vão voz passiva e enviam uma resposta ao r1, que vá voz passiva outra vez.

Se um problema é encontrado, a seguir um roteador pode ficar ativo por um tempo prolongado, porque deve esperar uma resposta. A fim impedir o roteador que espera uma resposta que possa nunca ser recebida, o roteador pode declarar o SIA e matar o neighborhood com que espera a resposta. A fim pesquisar defeitos o problema, para ver o **comando show ip eigrp topology active output** e segue a fuga do R.

Está aqui a saída para o r1 do roteador:

```
R1#show ip eigrp topology active
IP-EIGRP Topology Table for AS 1)/ID(10.100.1.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

A 10.100.1.1/32, 1 successors, FD is Inaccessible
  1 replies, active 00:01:11, query-origin: Local origin
    via Connected (Infinity/Infinity), Loopback0
    Remaining replies:
      via 10.1.1.2, r, Ethernet0/0
```

O r1 do roteador é ativo e espera uma resposta do R2. Está aqui a saída para o roteador R2:

```
R2#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(10.100.1.2)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
A 10.100.1.1/32, 1 successors, FD is Inaccessible
  1 replies, active 00:01:01, query-origin: Successor Origin
    via 10.1.1.1 (Infinity/Infinity), Ethernet0/0
    via 10.2.1.4 (Infinity/Infinity), r, Ethernet1/0, serno 524
    via 10.2.1.3 (Infinity/Infinity), Ethernet1/0, serno 523
```

O roteador R2 é ativo e espera uma resposta do R4. Está aqui a saída para o roteador R4:

```
R4#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(10.100.1.4)
```

```
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
A 10.100.1.1/32, 1 successors, FD is Inaccessible
  1 replies, active 00:00:56, query-origin: Successor Origin
    via 10.2.1.2 (Infinity/Infinity), Ethernet1/0
    via 172.16.1.5 (Infinity/Infinity), r, Serial2/0, serno 562
    via 10.2.1.3 (Infinity/Infinity), Ethernet1/0, serno 560
```

O roteador R4 é ativo e espera uma resposta do R5. Está aqui a saída para o roteador R5:

```
R5#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(172.16.1.5)
```

```
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
A 10.100.1.1/32, 1 successors, FD is Inaccessible, Q
  1 replies, active 00:00:53, query-origin: Successor Origin
    via 172.16.1.4 (Infinity/Infinity), Serial2/0
  Remaining replies:
    via 192.168.1.6, r, Serial3/0
```

O roteador R5 é ativo e espera uma resposta do R6. Está aqui a saída para o roteador R6:

```
R6#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(192.168.1.6)
R6#
```

Como mostrado, o roteador R6 não é ativo para o prefixo, assim que o problema deve estar entre o Roteadores R5 e R6. Após alguma hora, nós vemos que o R5 mata o neighborhood ao R6 e declara um estado SIA:

```
R6#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(192.168.1.6)
R6#
```

Quando você vê a saída para o roteador R5, você pode ver que há uns problemas no link para o R6.

Este é código novo SIA, e como tal, o SIA ocorreu em um roteador que fosse ao lado do problema. Neste exemplo, este é o link entre o Roteadores R5 e R6. Em umas versões de código mais velhas, o SIA poderia ser declarado em todo o roteador ao longo do trajeto (como no R2), que pôde ser distante do problema. O temporizador de SIA era três minutos. Todo o roteador ao longo do trajeto poderia ser o primeiro a ir SIA e matar os neighborhood. Com o código mais novo, o roteador espera uma resposta, envia intermediária uma pergunta SIA a seu vizinho, e as respostas do vizinho imediatamente com um SIA respondem. Por exemplo, quando no estado ativo, o roteador R4 enviar uma pergunta SIA ao R5, e as respostas R5 com um SIA respondem.

```
R6#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(192.168.1.6)
R6#
```

O roteador R5 igualmente envia perguntas SIA ao R6, mas não recebe uma resposta SIA do R6.

```
R6#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(192.168.1.6)
R6#
```

Uma vez que o roteador envia uma pergunta SIA mas não recebe uma resposta SIA, o **s** aparece para esse vizinho:

```
R5#show ip eigrp topology active
IP-EIGRP Topology Table for AS(1)/ID(172.16.1.5)
```

```
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
A 10.100.1.1/32, 1 successors, FD is Inaccessible, Qqr
  1 replies, active 00:02:36, query-origin: Successor Origin, retries(1)
    via 172.16.1.4 (Infinity/Infinity), Serial2/0, serno 61
    via 192.168.1.6 (Infinity/Infinity), rs, q, Serial3/0, serno 60, anchored
```

Com o código novo SIA, o SIA deve ser declarado no roteador R5 quando não recebe uma resposta SIA. Você deve então permitir a eliminação de erros para estes dois pacotes EIGRP SIA:

```
R2#debug eigrp packets SIAquery SIAreply
```

```
EIGRP Packets debugging is on
(SIAQUERY, SIAREPLY)
```

```
R2#show deb
```

```
EIGRP:
EIGRP Packets debugging is on
(SIAQUERY, SIAREPLY)
```

Em resumo, você pode usar estes comandos a fim pesquisar defeitos a edição SIA:

- **show ip eigrp topology ative**
- **mostre o evento do eigrp IP** (aumente possivelmente o tamanho do log de eventos)
- **mostre o tráfego do eigrp IP** (a busca para muitas perguntas SIA e respostas SIA)
- **mostre o mem do proc**
- **mostre a soma do mem**

Estão aqui algumas soluções possíveis para a edição SIA:

- Fixe o problema de link.
- Aplique a sumarização (manual ou automática) nas redes com muitos prefixos ou uma escala profunda da pergunta.
- Use distribuir-lista a fim diminuir a escala da pergunta.
- Defina roteadores remotos como topos.

## Prefixos faltantes

Há dois tipos de prefixos faltantes: aqueles que faltam na tabela de roteamento (ou no Routing Information Base (RIB)), e aqueles que faltam na tabela de topologia.

## Prefixos faltantes no RIB

Pode haver diversas razões que um prefixo não está incluído no RIB:

- O prefixo é instalado na tabela de roteamento por um outro protocolo de roteamento com uma distância administrativa mais baixa.
- Uma distribuir-lista obstrui o prefixo.
- Um horizonte dividido obstrui o prefixo.

## Prefixo instalado pelo protocolo de roteamento com distância administrativa mais baixa

Neste exemplo, o prefixo é instalado na tabela de roteamento por uma rota estática ou por um protocolo de roteamento com uma distância administrativa mais baixa.

Tipicamente quando isto ocorre, o prefixo está na tabela de topologia mas não tem nenhum sucessor. Você pode ver todas estas entradas com o comando dos **zero-sucessores da topologia do eigrp da mostra IP**. O feasible distance (FD) deve ter um valor infinito.

Incorpore o comando do **<prefix> da rota da mostra IP** e verifique os protocolos de roteamento esses próprios a rota no RIB:

```
R1#show ip eigrp topology 192.168.100.6 255.255.255.255
IP-EIGRP (AS 1): Topology entry for 192.168.100.6/32
  State is Passive, Query origin flag is 1, 0 Successor(s), FD is 4294967295
  Routing Descriptor Blocks:
  10.3.1.6 (Serial2/0), from 10.3.1.6, Send flag is 0x0
    Composite metric is (2297856/128256), Route is Internal
  Vector metric:
    Minimum bandwidth is 1544 Kbit
    Total delay is 25000 microseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 1R1#show ip eigrp topology zero-successors
IP-EIGRP Topology Table for AS(1)/ID(10.100.1.1)
```

```
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
P 192.168.1.0/24, 0 successors, FD is Inaccessible
  via 10.3.1.6 (2681856/2169856), Serial2/0
P 192.168.100.6/32, 0 successors, FD is Inaccessible
  via 10.3.1.6 (2297856/128256), Serial2/0
```

## A Distribuir-lista obstrui o prefixo

O EIGRP é um protocolo de roteamento de vetor de distância. Você pode usar uma distribuir-lista em todos os prefixos de bloco do roteador. Você pode usá-lo em uma relação a fim para os prefixos de ser mandado ou receba, ou você pode configurar a distribuir-lista globalmente sob o

processo de EIGRP do roteador a fim aplicar o filtro de roteamento em todas as relações EIGRP-permitidas.

Aqui está um exemplo:

```
R1#show running-config | begin router eigrp
```

```
router eigrp 1
network 10.0.0.0
distribute-list 1 in
no auto-summary
!
access-list 1 deny 192.168.100.6
access-list 1 permit any
```

## Prefixos faltantes na tabela de topologia

Esta seção descreve algumas das razões que um prefixo pôde faltar da tabela de topologia.

### Especificação da máscara para a saída apropriada do comando

Não faça o erro típico; quando você verifica um prefixo na tabela de topologia, especifique sempre a máscara. Isto ocorre se você não usa a máscara:

```
R1#show ip eigrp topology 192.168.100.6
% IP-EIGRP (AS 1): Route not in topology table
```

Está aqui o comando `show ip eigrp topology` output quando a máscara é especificada:

```
R1#show ip eigrp topology 192.168.100.6 255.255.255.255
IP-EIGRP (AS 1): Topology entry for 192.168.100.6/32
State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2297856
Routing Descriptor Blocks:
 10.3.1.6 (Serial2/0), from 10.3.1.6, Send flag is 0x0
   Composite metric is (2297856/128256), Route is Internal
   Vector metric:
     Minimum bandwidth is 1544 Kbit
     Total delay is 25000 microseconds
     Reliability is 255/255
     Load is 1/255
     Minimum MTU is 1500
     Hop count is 1
 10.4.1.5 (Ethernet1/0), from 10.4.1.5, Send flag is 0x
   Composite metric is (2323456/2297856), Route is Internal
   Vector metric:
     Minimum bandwidth is 1544 Kbit
     Total delay is 26000 microseconds
     Reliability is 255/255
     Load is 1/255
     Minimum MTU is 1500
     Hop count is 2
```

Como mostrado, o prefixo esta presente na tabela de topologia.

### O horizonte dividido obstrui o prefixo

Esta seção descreve um outro erro comum. O EIGRP não é um protocolo de roteamento de

estado de enlace, mas um pouco é um protocolo de roteamento de vetor de distância. A tabela de topologia deve ser usada para a operação correta do algoritmo de atualização Diffuse (DUPLO), não porque o EIGRP é um protocolo de roteamento de estado de enlace; daqui, exige um base de dados. A tabela de topologia é exigida porque somente as melhores rotas são instaladas na tabela de roteamento, quando as procuras DUPLAS que as rotas viáveis estão monitoradas também. Estes são armazenados na tabela de topologia.

Você deve sempre ter a rota do sucessor e as rotas viáveis na tabela de topologia. Se não, há um erro. Contudo, poderia igualmente haver rotas NON-praticáveis na tabela de topologia, enquanto são recebidos. Se não são recebidos de um vizinho, poderia haver um horizonte dividido que obstruísse o prefixo.

A saída do comando **show ip eigrp topology** mostra somente as entradas do prefixo que ponto aos sucessores e aos sucessores possíveis. Se você quer ver os prefixos que estão recebidos sobre todos os trajetos (também trajetos NON-praticáveis), a seguir inscreva o comando **show ip eigrp topology all-links** pelo contrário.

Aqui está um exemplo:

```
R1#show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.100.1.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.3.1.0/24, 1 successors, FD is 2169856
   via Connected, Serial2/0
P 10.2.1.0/24, 2 successors, FD is 307200
   via 10.1.1.2 (307200/281600), Ethernet0/0
   via 10.1.1.3 (307200/281600), Ethernet0/0
P 10.1.1.0/24, 1 successors, FD is 281600
   via Connected, Ethernet0/0
P 172.16.1.0/24, 1 successors, FD is 2195456
   via 10.4.1.5 (2195456/2169856), Ethernet1/0
P 192.168.1.0/24, 1 successors, FD is 2195456
   via 10.4.1.5 (2195456/2169856), Ethernet1/0
   via 10.3.1.6 (2681856/2169856), Serial2/0
P 10.4.1.0/24, 1 successors, FD is 281600
   via Connected, Ethernet1/0
P 172.16.100.5/32, 1 successors, FD is 409600
   via 10.4.1.5 (409600/128256), Ethernet1/0
P 10.100.1.4/32, 2 successors, FD is 435200
   via 10.1.1.2 (435200/409600), Ethernet0/0
   via 10.1.1.3 (435200/409600), Ethernet0/0
P 10.100.1.3/32, 1 successors, FD is 409600
   via 10.1.1.3 (409600/128256), Ethernet0/0
P 10.100.1.2/32, 1 successors, FD is 409600
   via 10.1.1.2 (409600/128256), Ethernet0/0
P 10.100.1.1/32, 1 successors, FD is 128256
   via Connected, Loopback0
P 192.168.100.6/32, 1 successors, FD is 2297856
   via 10.3.1.6 (2297856/128256), Serial2/0
```

Nesta saída você pode ver que a parcela dos **todo-links** do comando inclui mais trajetos:

```
R1#show ip eigrp topology all-links
IP-EIGRP Topology Table for AS(1)/ID(10.100.1.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```

P 10.3.1.0/24, 1 successors, FD is 2169856, serno 43
  via Connected, Serial2/0
P 10.2.1.0/24, 2 successors, FD is 307200, serno 127
  via 10.1.1.2 (307200/281600), Ethernet0/0
  via 10.1.1.3 (307200/281600), Ethernet0/0
P 10.1.1.0/24, 1 successors, FD is 281600, serno 80
  via Connected, Ethernet0/0
P 172.16.1.0/24, 1 successors, FD is 2195456, serno 116
  via 10.4.1.5 (2195456/2169856), Ethernet1/0
  via 10.3.1.6 (3193856/2681856), Serial2/0
  via 10.1.1.2 (2221056/2195456), Ethernet0/0
  via 10.1.1.3 (2221056/2195456), Ethernet0/0
P 192.168.1.0/24, 1 successors, FD is 2195456, serno 118
  via 10.4.1.5 (2195456/2169856), Ethernet1/0
  via 10.3.1.6 (2681856/2169856), Serial2/0
P 10.4.1.0/24, 1 successors, FD is 281600, serno 70
  via Connected, Ethernet1/0
P 172.16.100.5/32, 1 successors, FD is 409600, serno 117
  via 10.4.1.5 (409600/128256), Ethernet1/0
  via 10.3.1.6 (2809856/2297856), Serial2/0
P 10.100.1.4/32, 2 successors, FD is 435200, serno 128
  via 10.1.1.2 (435200/409600), Ethernet0/0
  via 10.1.1.3 (435200/409600), Ethernet0/0
P 10.100.1.3/32, 1 successors, FD is 409600, serno 115
  via 10.1.1.3 (409600/128256), Ethernet0/0
P 10.100.1.2/32, 1 successors, FD is 409600, serno 109
  via 10.1.1.2 (409600/128256), Ethernet0/0
P 10.100.1.1/32, 1 successors, FD is 128256, serno 4
  via Connected, Loopback0
P 192.168.100.6/32, 1 successors, FD is 2297856, serno 135
  via 10.3.1.6 (2297856/128256), Serial2/0
  via 10.4.1.5 (2323456/2297856), Ethernet1/0

```

Considere o último prefixo na saída precedente; o trajeto através de **10.4.1.5** tem **(2323456/2297856)**. A distância informada (métrica anunciada) é **2297856**, que não é menor do que o FD de **2297856**, assim que o trajeto não é praticável.

```

P 192.168.100.6/32, 1 successors, FD is 2297856, serno 135
  via 10.3.1.6 (2297856/128256), Serial2/0
  via 10.4.1.5 (2323456/2297856), Ethernet1/0

```

Está aqui um exemplo onde um horizonte dividido faz com que um trajeto seja excluído da tabela de topologia para uma rota. Quando você vê a topologia, você pode ver que o r1 do roteador tem o prefixo **192.168.100.6/32** através do R6 e do R5 na tabela de topologia, mas não através do R2 ou do R3:

```

R1#show ip eigrp topology 192.168.100.6 255.255.255.255
IP-EIGRP (AS 1): Topology entry for 192.168.100.6/32
State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2297856
Routing Descriptor Blocks:
10.3.1.6 (Serial2/0), from 10.3.1.6, Send flag is 0x0
  Composite metric is (2297856/128256), Route is Internal
  Vector metric:
    Minimum bandwidth is 1544 Kbit
    Total delay is 25000 microseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 1
10.4.1.5 (Ethernet1/0), from 10.4.1.5, Send flag is 0x0
  Composite metric is (2323456/2297856), Route is Internal
  Vector metric:
    Minimum bandwidth is 1544 Kbit

```



```
Total delay is 26000 microseconds
Reliability is 255/255
Load is 1/255
Minimum MTU is 1500
Hop count is 2
```

Isto é porque o r1 do roteador nunca recebeu o prefixo 192.168.100.6/32 através do R2 ou do R3, porque têm o prefixo 192.168.100.6/32 através do r1 na tabela de roteamento.

```
R2#show ip route 192.186.100.6 255.255.255.255
Routing entry for 192.168.100.6/32
  Known via "eigrp 1", distance 90, metric 2323456, type internal
  Redistributing via eigrp 1
  Last update from 10.1.1.1 on Ethernet0/0, 00:02:07 ago
  Routing Descriptor Blocks:
  * 10.1.1.1, from 10.1.1.1, 00:02:07 ago, via Ethernet0/0
    Route metric is 2323456, traffic share count is 1
    Total delay is 26000 microseconds, minimum bandwidth is 1544 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 2
```

```
R3#show ip route 192.168.100.6 255.255.255.255
Routing entry for 192.168.100.6/32
  Known via "eigrp 1", distance 90, metric 2323456, type internal
  Redistributing via eigrp 1
  Last update from 10.1.1.1 on Ethernet0/0, 00:01:58 ago
  Routing Descriptor Blocks:
  * 10.1.1.1, from 10.1.1.1, 00:01:58 ago, via Ethernet0/0
    Route metric is 2323456, traffic share count is 1
    Total delay is 26000 microseconds, minimum bandwidth is 1544 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 2
```

A fim verificar isto, use a palavra-chave dos **todo-links no r1** quando você vê a tabela de topologia. Isto mostra todos os trajetos para todos os prefixos, que inclui os trajetos NON-praticáveis. Você pode então ver que o prefixo 192.168.100.6/32 não esteve aprendido pelo r1 do roteador do R2 ou do R3.

## Métrica

Nota: O MTU e o contagem de saltos não são incluídos no cálculo métrico.

Estas são as fórmulas que são usadas a fim calcular a métrica do trajeto de uma rota:

- Se o K5 é um valor diferente de zero:

$$\text{Métrica EIGRP} = 256 * (((K1 * Bw) + (K2 * Bw) / (256 - Load) + (K3 * Delay)) * (K5 / (Reliability + K4)))$$

- Se o K5 é igual a zero:

$$\text{Métrica EIGRP} = 256 * ((K1 * Bw) + (K2 * Bw) / (256 - Load) + (K3 * Delay))$$

Os valores K estão a uns pesos que sejam usados a fim tornar mais pesados os quatro componentes da métrica EIGRP: atraso, largura de banda, confiança, e carga. Estes são os valores K do padrão:

- K1 = 1
- K2 = 0
- K3 = 1
- K4 = 0
- K5 = 0

Com os valores K do padrão (somente usando a largura de banda e o atraso), a fórmula torna-se:

**Métrica EIGRP = 256 \* (BW + atraso)**

**Bw= (BW 10<sup>7</sup>/minimum nos kilobits por segundo)**

Nota: O atraso é medido nas dezenas de microssegundos; contudo, na relação, é medido nos microssegundos.

Todos os quatro componentes podem ser verificados com o **comando show interface**:

```
R1#show interface et 0/0
Ethernet0/0 is up, line protocol is up
Hardware is AmdP2, address is aabb.cc00.0100 (bia aabb.cc00.0100)
Internet address is 10.1.1.1/24
MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set  Keepalive set (10 sec)
ARP type: ARPA, ARP Timeout 04:00:00  Last input 00:00:02, output 00:00:02,
output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  789 packets input, 76700 bytes, 0 no buffer
  Received 707 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
  0 input packets with dribble condition detected
  548 packets output, 49206 bytes, 0 underruns
  0 output errors, 0 collisions, 1 interface resets
  0 unknown protocol drops
  0 babbles, 0 late collision, 0 deferred
  0 lost carrier, 0 no carrier
  0 output buffer failures, 0 output buffers swapped out
```

O atraso é cumulativo, assim que significa que você adiciona o atraso de cada link ao longo do trajeto. A largura de banda não é cumulativa, assim a largura de banda que é usada na fórmula é a largura de banda a menor de todo o link ao longo do trajeto.

## ID do roteador duplicado

A fim ver o Router ID que os usos EIGRP, inscrevem o **comando show ip eigrp topology** no roteador e veem a primeira linha da saída:

```
R1#show ip eigrp topology
```

```
IP-EIGRP Topology Table for AS(1)/ID(10.100.1.1)
```

```
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,  
r - reply Status, s - sia Status
```

```
P 10.3.1.0/24, 1 successors, FD is 2169856  
via Connected, Serial2/0
```

O EIGRP Router ID não é usado de todo para rotas internas em umas versões do Cisco IOS mais velhas. Um ID do roteador duplicado para o EIGRP não deve causar nenhuns problemas se somente as rotas internas são usadas. Em um Cisco IOS Software mais novo, as rotas internas EIGRP levam o EIGRP Router ID.

O Router ID para rotas externas pode ser visto nesta saída:

```
R1#show ip eigrp topology 192.168.1.4 255.255.255.255
```

```
IP-EIGRP (AS 1): Topology entry for 192.168.1.4/32
```

```
State is Passive, Query origin flag is 1, 2 Successor(s), FD is 435200
```

```
Routing Descriptor Blocks:
```

```
10.1.1.2 (Ethernet0/0), from 10.1.1.2, Send flag is 0x0
```

```
Composite metric is (435200/409600), Route is External
```

```
Vector metric:
```

```
Minimum bandwidth is 10000 Kbit
```

```
Total delay is 7000 microseconds
```

```
Reliability is 255/255
```

```
Load is 1/255
```

```
Minimum MTU is 1500
```

```
Hop count is 2
```

```
External data&colon;
```

```
Originating router is 10.100.1.4
```

```
AS number of route is 0
```

```
External protocol is Connected, external metric is 0
```

```
Administrator tag is 0 (0x00000000)
```

Se uma rota de EIGRP (externo) com o mesmo EIGRP Router ID que o roteador é recebida, não gerencie uma entrada de registro. Contudo, o log de eventos EIGRP captura este. Quando você verifica para ver se há a rota de EIGRP (externo), não aparece na tabela de topologia.

Verifique o log de eventos EIGRP para ver se há mensagens possíveis de ID do roteador duplicado:

```
R1#show ip eigrp events
```

```
Event information for AS 1:
```

```
1 08:36:35.303 Ignored route, metric: 10.33.33.33 3347456  
2 08:36:35.303 Ignored route, neighbor info: 10.3.1.6 Serial2/1  
3 08:36:35.303 Ignored route, dup router: 10.100.1.1  
4 08:36:35.303 Rcv EOT update src/seq: 10.3.1.6 143  
5 08:36:35.227 Change queue emptied, entries: 2  
6 08:36:35.227 Route OBE net/refcount: 10.100.1.4/32 3  
7 08:36:35.227 Route OBE net/refcount: 10.2.1.0/24 3  
8 08:36:35.227 Metric set: 10.100.1.4/32 435200  
9 08:36:35.227 Update reason, delay: nexthop changed 179200  
10 08:36:35.227 Update sent, RD: 10.100.1.4/32 435200  
11 08:36:35.227 Route install: 10.100.1.4/32 10.1.1.3  
12 08:36:35.227 Route install: 10.100.1.4/32 10.1.1.2  
13 08:36:35.227 RDB delete: 10.100.1.4/32 10.3.1.6
```

## Os valores K combinam mal/paradas oportunas

Quando os valores K não são os mesmos em roteadores vizinho, esta mensagem está

observada:

```
R1#show ip eigrp events
```

```
Event information for AS 1:
```

```
1 08:36:35.303 Ignored route, metric: 10.33.33.33 3347456
2 08:36:35.303 Ignored route, neighbor info: 10.3.1.6 Serial2/1
3 08:36:35.303 Ignored route, dup router: 10.100.1.1
4 08:36:35.303 Rcv EOT update src/seq: 10.3.1.6 143
5 08:36:35.227 Change queue emptied, entries: 2
6 08:36:35.227 Route OBE net/refcount: 10.100.1.4/32 3
7 08:36:35.227 Route OBE net/refcount: 10.2.1.0/24 3
8 08:36:35.227 Metric set: 10.100.1.4/32 435200
9 08:36:35.227 Update reason, delay: nexthop changed 179200
10 08:36:35.227 Update sent, RD: 10.100.1.4/32 435200
11 08:36:35.227 Route install: 10.100.1.4/32 10.1.1.3
12 08:36:35.227 Route install: 10.100.1.4/32 10.1.1.2
13 08:36:35.227 RDB delete: 10.100.1.4/32 10.3.1.6
```

Os valores K são configurados com este comando (com os valores possíveis de K entre 0 e 255):

```
metric weights tos k1 k2 k3 k4 k5
```

```
!
router eigrp 1
network 10.0.0.0
metric weights 0 1 2 3 4 5
!
```

A mensagem indica que a vizinhança do EIGRP não está estabelecida devido a uma má combinação nos valores K. Os valores K devem ser os mesmos em todos os EIGRP Router em um sistema autônomo a fim impedir problemas de roteamento quando o Roteadores diferente usa cálculos da métrica diferente.

Verifique se os valores K sejam os mesmos nos roteadores vizinho. Se os valores K são os mesmos, a edição pôde ser causada pela característica da parada oportuna EIGRP. Nesse caso, um roteador envia um pacote de hello de EIGRP com os valores K ajustados a 255 de modo que a má combinação dos valores K ocorra intencionalmente. Esta é indicar ao EIGRP Router vizinho que está indo para baixo. No roteador vizinho, você veria este *mensagem de saída* recebido:

```
metric weights tos k1 k2 k3 k4 k5
```

```
!
router eigrp 1
network 10.0.0.0
metric weights 0 1 2 3 4 5
!
```

Contudo, se o roteador vizinho executa uma versão de código mais velha (antes da identificação de bug Cisco [CSCdr96531](#)), não reconhece este como uma mensagem da parada oportuna, mas como uma má combinação nos valores K:

```
metric weights tos k1 k2 k3 k4 k5
```

```
!
router eigrp 1
network 10.0.0.0
metric weights 0 1 2 3 4 5
!
```

Esta é a mesma mensagem que no caso de uma má combinação verdadeira dos valores K nos roteadores vizinho.

Estes são os disparadores para uma parada oportuna:

- Nenhum comando `router eigrp` é inscrito.
- Nenhum comando `network` é inscrito.
- O comando `ip eigrp neighbor` claro é inscrito.
- O roteador é recarregado.

Uma parada oportuna é usada a fim acelerar a detecção de um estado inativo vizinho. Sem uma parada oportuna, um vizinho deve esperar até que o tempo de contenção expire antes que declare o vizinho estar para baixo.

## Balanceamento de carga dos custos desiguais (variação)

O Balanceamento de carga dos custos desiguais é possível no EIGRP com o comando `variance`, mas a variação e as condições de viabilidade devem ser estadas conformes.

A condição da variação significa que a métrica da rota não é maior do que a melhor métrica multiplicada pela variação. Para que uma rota seja considerada praticável, a rota deve ter sido anunciada com uma distância informada que seja mais baixa do que o feasible distance (FD). Aqui está um exemplo:

```
!
router eigrp 1
variance 2
network 10.0.0.0
no auto-summary
!
```

O r1 do roteador tem uma **variação 2** configurada. Isto significa que se o roteador tem um outro trajeto para a rota com uma métrica que não seja maior de duas vezes a melhor métrica para essa rota, deve haver Balanceamento de carga dos custos desiguais para essa rota.

```
R1#show ip eigrp topology 172.16.100.5 255.255.255.255
IP-EIGRP (AS 1): Topology entry for 172.16.100.5/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 409600
  Routing Descriptor Blocks:
  10.4.1.5 (Ethernet1/0), from 10.4.1.5, Send flag is 0x0
    Composite metric is (409600/128256), Route is Internal
    Vector metric:
      Minimum bandwidth is 10000 Kbit
      Total delay is 6000 microseconds
      Reliability is 255/255
      Load is 1/255
      Minimum MTU is 1500
      Hop count is 1
  10.3.1.6 (Serial2/0), from 10.3.1.6, Send flag is 0x0
    Composite metric is (435200/409600), Route is Internal <<< RD = 409600
    Vector metric:
      Minimum bandwidth is 10000 Kbit
      Total delay is 7000 microseconds
      Reliability is 255/255
      Load is 1/255
      Minimum MTU is 1500
      Hop count is 2
```

Se a segunda entrada da topologia é instalada na tabela de roteamento, a métrica da segunda entrada da topologia é **435200**. Desde que duas vezes a melhor métrica é  $2 \times 409600 = 819200$ ,

e 435200 < 819200, a segunda entrada da topologia está dentro da escala da variação. A distância informada da segunda entrada da topologia é 409600, que não é menor do que FD = 409600. A segunda circunstância (possibilidade) não é estada conforme, e a segunda entrada não pode ser instalada no RIB.

```
R1#show ip route 172.16.100.5
Routing entry for 172.16.100.5/32
  Known via "eigrp 1", distance 90, metric 409600, type internal
  Redistributing via eigrp 1
  Last update from 10.4.1.5 on Ethernet1/0, 00:00:16 ago
  Routing Descriptor Blocks:
  * 10.4.1.5, from 10.4.1.5, 00:00:16 ago, via Ethernet1/0
    Route metric is 409600, traffic share count is 1
    Total delay is 6000 microseconds, minimum bandwidth is 10000 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 1
```

Se o RD da segunda entrada da topologia é menor então o FD, como no exemplo seguinte, haveria Balanceamento de carga dos custos desiguais.

```
R1#show ip eigrp topology 172.16.100.5 255.255.255.255
IP-EIGRP (AS 1): Topology entry for 172.16.100.5/32
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 409600
  Routing Descriptor Blocks:
  10.4.1.5 (Ethernet1/0), from 10.4.1.5, Send flag is 0x0
    Composite metric is (409600/128256), Route is Internal
    Vector metric:
      Minimum bandwidth is 10000 Kbit
      Total delay is 6000 microseconds
      Reliability is 255/255
      Load is 1/255
      Minimum MTU is 1500
      Hop count is 1
  10.3.1.6 (Serial2/0), from 10.3.1.6, Send flag is 0x0
    Composite metric is (434944/409344), Route is Internal <<< RD = 409344
    Vector metric:
      Minimum bandwidth is 10000 Kbit
      Total delay is 6990 microseconds
      Reliability is 255/255
      Load is 1/255
      Minimum MTU is 1500
      Hop count is 2
```

Ambas as entradas da topologia estão agora na tabela de roteamento:

```
R1#show ip route 172.16.100.5
Routing entry for 172.16.100.5/32
  Known via "eigrp 1", distance 90, metric 409600, type internal
  Redistributing via eigrp 1
  Last update from 10.3.1.6 on Serial2/0, 00:00:26 ago
  Routing Descriptor Blocks:
  * 10.4.1.5, from 10.4.1.5, 00:00:26 ago, via Ethernet1/0
    Route metric is 409600, traffic share count is 120
    Total delay is 6000 microseconds, minimum bandwidth is 10000 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 1
  10.3.1.6, from 10.3.1.6, 00:00:26 ago, via Serial2/0
    Route metric is 434944, traffic share count is 113
    Total delay is 6990 microseconds, minimum bandwidth is 10000 Kbit
    Reliability 255/255, minimum MTU 1500 bytes
    Loading 1/255, Hops 2
```

# Vizinhos estáticos

O EIGRP apoia configurações com uns ou vários vizinhos estáticos na mesma relação. Assim que você configurar um vizinho EIGRP estático na relação, o roteador já não envia os pacotes EIGRP como o Multicast nessa relação ou processa os pacotes EIGRP multicasted recebidos. Isto significa que os pacotes olá!, da atualização, e da pergunta unicasted agora. Nenhum neighborships adicional pode ser formado a menos que o comando do **vizinho estático** for configurado explicitamente para aqueles vizinhos nessa relação.

Isto é como configurar um vizinho EIGRP estático:

```
router eigrp 1
passive-interface Loopback0
network 10.0.0.0
no auto-summary
neighbor 10.1.1.1 Ethernet0/0
!
```

Quando o Roteadores em ambos os lados do link tem o comando do **vizinho estático**, o neighborship está formado:

```
R1#show ip eigrp neighbors detail
IP-EIGRP neighbors for process 1
H   Address                Interface          Hold Uptime    SRTT   RTO  Q  Seq
                               (sec)           (ms)          Cnt Num
1   10.1.1.2                Et0/0             14 00:00:23   27    200  0  230
  Static neighbor
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 1
0   10.3.1.6                Se2/0             14 1d02h      26    200  0  169
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 12
3   10.4.1.5                Et1/0             10 1d02h      16    200  0  234
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 7
```

Se somente um roteador tem o comando do **vizinho estático** configurado, você observará que o roteador ignora os pacotes EIGRP multicasted e o outro roteador ignora os pacotes EIGRP unicasted:

```
R1#show ip eigrp neighbors detail
IP-EIGRP neighbors for process 1
H   Address                Interface          Hold Uptime    SRTT   RTO  Q  Seq
                               (sec)           (ms)          Cnt Num
1   10.1.1.2                Et0/0             14 00:00:23   27    200  0  230
  Static neighbor
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 1
0   10.3.1.6                Se2/0             14 1d02h      26    200  0  169
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 12
3   10.4.1.5                Et1/0             10 1d02h      16    200  0  234
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 7R1#show ip eigrp neighbors detail
IP-EIGRP neighbors for process 1
H   Address                Interface          Hold Uptime    SRTT   RTO  Q  Seq
                               (sec)           (ms)          Cnt Num
1   10.1.1.2                Et0/0             14 00:00:23   27    200  0  230
  Static neighbor
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 1
0   10.3.1.6                Se2/0             14 1d02h      26    200  0  169
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 12
3   10.4.1.5                Et1/0             10 1d02h      16    200  0  234
  Version 12.4/1.2, Retrans: 0, Retries: 0, Prefixes: 7
```

Há um comando debug especial para vizinhos estáticos EIGRP:

```
R2#debug eigrp neighbors static
EIGRP Static Neighbors debugging is on
```

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#router eigrp 1
R2(config-router)#neighbor 10.1.1.1 et 0/0
R2(config-router)#end
R2#
```

```
EIGRP: Multicast Hello is disabled on Ethernet0/0!
EIGRP: Add new static nbr 10.1.1.1 to AS 1 Ethernet0/0
```

Estão aqui os alguns motivos que os vizinhos EIGRP estáticos puderam ser configurados:

- Você quer limitar ou evitar transmissões em redes do Non-Broadcast Multi-Access (NBMA).
- Você quer limitar ou evitar Multicast em meios de transmissão (Ethernet).
- Para os propósitos de Troubleshooting (que usam o unicast em vez do Multicast).

Cuidado: Não configurar o **comando passive-interface** junto com o comando **estático do vizinho EIGRP**.

## Redistribuição da rota estática

Quando você configura uma rota estática que aponte a uma relação, e à rota é coberto por uma instrução de rede sob o roteador EIGRP, a seguir a rota estática está anunciada pelo EIGRP como se era uma rota conectada. O **comando redistribute static** ou uma métrica do padrão não são exigidos neste caso.

```
R2#debug eigrp neighbors static
EIGRP Static Neighbors debugging is on
```

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#router eigrp 1
R2(config-router)#neighbor 10.1.1.1 et 0/0
R2(config-router)#end
R2#
```

```
EIGRP: Multicast Hello is disabled on Ethernet0/0!
EIGRP: Add new static nbr 10.1.1.1 to AS 1 Ethernet0/0R1#show ip eigrp top 172.16.0.0
255.255.0.0
```

```
IP-EIGRP (AS 1): Topology entry for 172.16.0.0/16
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2169856
  Routing Descriptor Blocks:
  0.0.0.0, from Rstatic, Send flag is 0x0
    Composite metric is (2169856/0), Route is Internal
    Vector metric:
      Minimum bandwidth is 1544 Kbit
      Total delay is 20000 microseconds
      Reliability is 255/255
      Load is 1/255
      Minimum MTU is 1500
      Hop count is 0
```



# Confiança e carga para o cálculo métrico

Cuidado: A confiança e/ou a carga de uso calculam o medidor.

A confiança e os parâmetros de carga aparecem na saída do **comando show interface**. Não há nenhuma atualização dinâmica para estes parâmetros quando a carga e a confiança mudam. Se a carga e a confiança mudam, não provoca uma mudança imediata na métrica. Somente se o EIGRP decide enviar as atualizações a seus vizinhos devido às alterações de topologia querem a mudança na carga e a confiança seja propagada. Além disso, o uso da carga e da confiança a fim calcular a métrica pode introduzir a instabilidade, porque o roteamento adaptativo é executado então. Se você deseja mudar o roteamento de acordo com a carga de tráfego, a seguir você deve considerar o uso da engenharia de tráfego do Multiprotocol Label Switching (MPLS) ou do roteamento do desempenho (PFR).

## Alta utilização da CPU

Há três processos de EIGRP que são executado simultaneamente:

- O do **roteador** este processo guarda as associações de memória compartilhada.
- **Olá!** que do este processo envia e recebe os pacotes Hello e mantém as conexões de peer.
- O do **módulo dependente de protocolo (PDM)** O EIGRP apoia quatro conjuntos de protocolos: IP, IPv6, IPX, e APPLETALK. Cada série tem seu próprio PDM. Estão aqui as funções principal do PDM:

Mantém o vizinho e as tabelas de topologia dos EIGRP Router que pertencem a esse conjunto de protocolos.

As construções e traduzem pacotes do específico de protocolo para DUPLIO (transmissão e recepção dos pacotes EIGRP).

As relações DUAL à tabela de roteamento do específico de protocolo.

Computa a métrica e passam a informação PARA DUAL (DUAL somente as picaretas os sucessores e os sucessores possíveis).

Filtração e listas de acesso dos implementares.

Executa funções da redistribuição a e dos outros protocolos de roteamento.

Estão aqui umas saídas de exemplo que mostrem estes três processos:

```
R1#show proc cpu | include EIGRP
89          4          24          166 0.00% 0.00% 0.00% 0 IP-EIGRP Router
90         1016         4406          230 0.00% 0.03% 0.00% 0 IP-EIGRP: PDM
91         2472         6881          359 0.00% 0.07% 0.08% 0 IP-EIGRP: HELLO
```

A alta utilização da CPU no EIGRP não é normal. Se isto ocorre, ou o EIGRP tem demasiado a fazer ou há um erro no EIGRP. No primeiro caso, verifique o número de prefixos na tabela de topologia e o número de pares. Verifique para ver se há a instabilidade entre as rotas de EIGRP e os vizinhos.

## EIGRP nas redes do Frame Relay (fila de broadcast)

Nas redes do Frame Relay onde há Roteadores do vizinho múltiplo em uma interface ponto a multiponto, lá pode estar muito a transmissão ou os pacotes de transmissão múltipla que devem ser transmitidos. Por este motivo, há uma fila de broadcast separada com seus próprios buffers. A fila de broadcast tem a prioridade quando transmite em uma taxa abaixo da máxima configurada e tem uma alocação de largura de banda mínima garantida.

Está aqui o comando que é usado nesta encenação:

```
frame-relay broadcast-queue size byte-rate packet-rate
```

Em regra geral, comece com os vinte pacotes pelo identificador da conexão de link de dados (DLCI). A taxa de byte deve ser menos do que ambos:

- O  $N/4$  cronometra a taxa de Acesso remoto mínima (medida nos bytes por segundo), onde N é o número de DLCI a que a transmissão deve ser replicated.
- Um quarto da taxa de acesso local (medida nos bytes por segundo).

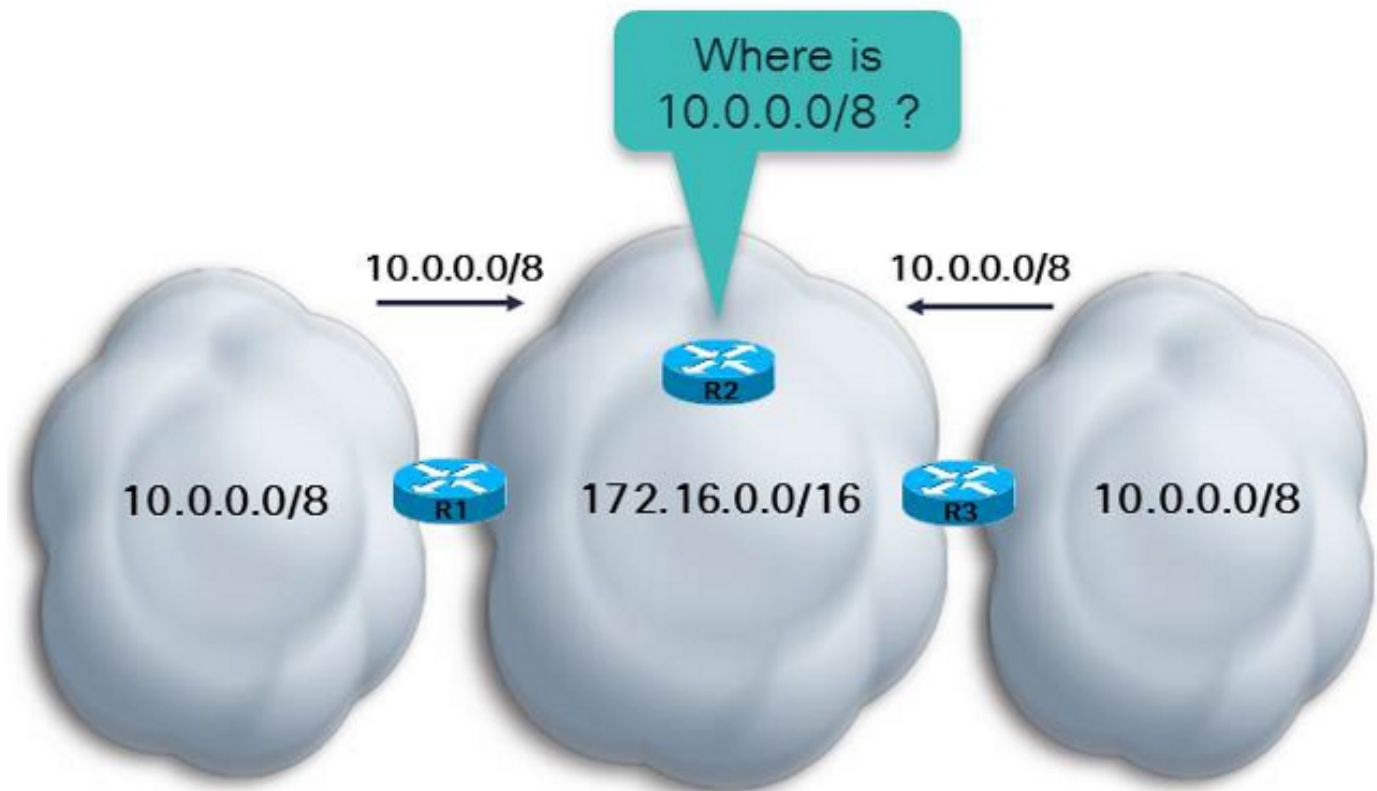
Se você observa um grande número vizinhos EIGRP bater, aumente o tamanho da fila de broadcast do Frame Relay. Esta edição não está atual se há umas subinterfaces do Frame Relay porque cada roteador vizinho está em uma subinterface com uma sub-rede diferente IP. Considere isto como uma ação alternativa quando há uma grande, rede do Frame Relay malha cheia.

## Combinado mal COMO números

Quando você incorpora o comando dos **pacotes EIGRP debug olá!**, revela que o roteador não recebe os pacotes Hello.

## Resumo automático

O EIGRP usado para executar a sumarização na rede principal (redes A, B, e C) limites à revelia. Isto significa que umas rotas mais específicas do que /8 prefixam para o tipo A da rede principal, umas rotas mais específicas do que os prefixos de /16 para o tipo B da rede principal, e umas rotas mais específicas do que os prefixos de /24 para redes principal datilogram o C, são perdidos quando cruzam seus limites. Está aqui um exemplo onde o resumo automático causa um problema:



Como mostrado, o R1 dos roteadores e o R3 têm o *resumo automático* sob o roteador EIGRP. O roteador R2 recebe 10.0.0.0/8 dos roteadores R1 e do R3 porque o R1 e o R3 são roteadores limítrofes entre a rede 10.0.0.0/8 e 172.16.0.0/16 da classe principal A. O roteador R2 pode ter a rota 10.0.0.0/8 através do R1 e do R3 se a métrica acontecer ser a mesma. Se não, o R2 tem a rota 10.0.0.0/8 através do R1 ou através do R3, dependente do trajeto que produz menos custo. Em qualquer dos casos, se o R2 deve enviar o tráfego a determinadas sub-redes de 10.0.0.0/8, não pode ser completamente certo que o tráfego alcança seu destino, como uma sub-rede de 10.0.0.0/8 pode estar somente no nuvem de rede esquerdo ou direito.

A fim aliviar este problema, não datilografe simplesmente **nenhum resumo automático** sob o processo de EIGRP do roteador. O roteador propaga então sub-redes das redes principais através do limite. Em umas versões do Cisco IOS mais novas, *nenhum* ajuste do *resumo automático* é o comportamento padrão.

## Log de eventos EIGRP

O log de eventos EIGRP captura os eventos EIGRP. É similar a quando debug são permitidos para o EIGRP. Contudo, é menos disruptivo e é executado à revelia. Pode ser usado a fim capturar os eventos que são mais difíceis de pesquisar defeitos ou uns eventos mais intermitentes. Este log é à revelia somente 500 linhas. A fim aumentá-lo, incorpore o comando do **209878>** do **^ do ^ do evento-log-tamanho <0 do eigrp**. Você pode aumentar o tamanho do log tanto quanto desejado, mas mantém na mente a quantidade de memória que o roteador tem que poupar para este log. A fim cancelar o log de eventos EIGRP, incorpore o comando **claro dos eventos do eigrp IP**.

Aqui está um exemplo:

```
R1#show ip eigrp events
Event information for AS 1:
```

```
1 09:01:36.107 Poison squashed: 10.100.1.3/32 reverse
2 09:01:35.991 Update ACK: 10.100.1.4/32 Serial2/0
3 09:01:35.967 Update ACK: 10.100.1.4/32 Ethernet0/0
4 09:01:35.967 Update ACK: 10.100.1.4/32 Ethernet1/0
5 09:01:35.943 Update delay/poison: 179200 FALSE
6 09:01:35.943 Update transmitted: 10.100.1.4/32 Serial2/0
7 09:01:35.943 Update delay/poison: 179200 TRUE
8 09:01:35.943 Update transmitted: 10.100.1.4/32 Ethernet0/0
9 09:01:35.943 Update delay/poison: 179200 FALSE
10 09:01:35.943 Update transmitted: 10.100.1.4/32 Ethernet1/0
11 09:01:35.923 Update packetized: 10.100.1.4/32 Ethernet0/0
12 09:01:35.923 Update packetized: 10.100.1.4/32 Ethernet1/0
13 09:01:35.923 Update packetized: 10.100.1.4/32 Serial2/0
14 09:01:35.903 Change queue emptied, entries: 1
15 09:01:35.903 Route OBE net/refcount: 10.100.1.4/32 3
16 09:01:35.903 Metric set: 172.16.1.0/24 2195456
17 09:01:35.903 Route install: 172.16.1.0/24 10.4.1.5
18 09:01:35.903 FC sat rdbmet/succmet: 2195456 2169856
19 09:01:35.903 FC sat nh/ndbmet: 10.4.1.5 2195456
20 09:01:35.903 Find FS: 172.16.1.0/24 2195456
```

A maioria de acontecimentos recentes aparecem na parte superior do log. Você pode filtrar determinados tipos de eventos EIGRP, tais como DUPLO, Xmit, e transporte:

```
R1#show ip eigrp events
```

```
Event information for AS 1:
```

```
1 09:01:36.107 Poison squashed: 10.100.1.3/32 reverse
2 09:01:35.991 Update ACK: 10.100.1.4/32 Serial2/0
3 09:01:35.967 Update ACK: 10.100.1.4/32 Ethernet0/0
4 09:01:35.967 Update ACK: 10.100.1.4/32 Ethernet1/0
5 09:01:35.943 Update delay/poison: 179200 FALSE
6 09:01:35.943 Update transmitted: 10.100.1.4/32 Serial2/0
7 09:01:35.943 Update delay/poison: 179200 TRUE
8 09:01:35.943 Update transmitted: 10.100.1.4/32 Ethernet0/0
9 09:01:35.943 Update delay/poison: 179200 FALSE
10 09:01:35.943 Update transmitted: 10.100.1.4/32 Ethernet1/0
11 09:01:35.923 Update packetized: 10.100.1.4/32 Ethernet0/0
12 09:01:35.923 Update packetized: 10.100.1.4/32 Ethernet1/0
13 09:01:35.923 Update packetized: 10.100.1.4/32 Serial2/0
14 09:01:35.903 Change queue emptied, entries: 1
15 09:01:35.903 Route OBE net/refcount: 10.100.1.4/32 3
16 09:01:35.903 Metric set: 172.16.1.0/24 2195456
17 09:01:35.903 Route install: 172.16.1.0/24 10.4.1.5
18 09:01:35.903 FC sat rdbmet/succmet: 2195456 2169856
19 09:01:35.903 FC sat nh/ndbmet: 10.4.1.5 2195456
20 09:01:35.903 Find FS: 172.16.1.0/24 2195456
```

Adicionalmente, você pode permitir o registro para um destes três tipos, uma combinação de dois tipos, ou para todos os três. Está aqui um exemplo onde dois tipos de registro sejam permitidos:

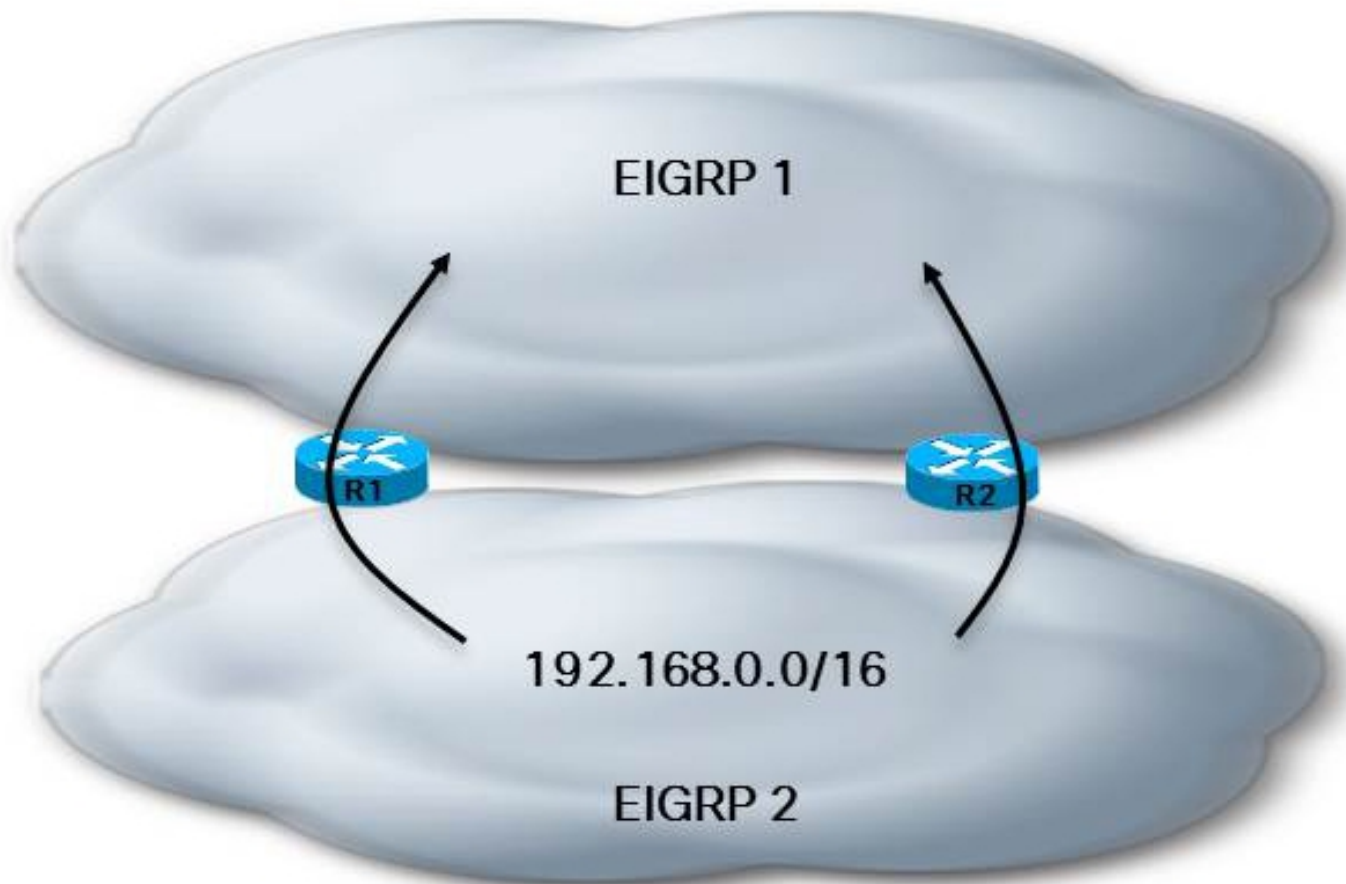
```
router eigrp 1
 redistribute connected
 network 10.0.0.0
 no auto-summary
 eigrp log-event-type dual xmit
 eigrp event-logging
 eigrp event-log-size 100000
!
```

Cuidado: Quando você permite o **logging de evento do eigrp**, imprime o logging de evento e armazena-o na tabela de evento. Isto pode conduzir a uma grande quantidade de saída impressa no console, similar a quando a eliminação de erros pesada EIGRP é permitida.

# A mesma rede aprendida por dois sistemas autônomo de EIGRP

Se uma rota é instruída através de dois processos de EIGRP, a seguir somente um dos processos de EIGRP pode instalar a rota no RIB. O processo com a distância administrativa mais baixa instala a rota. Se a distância administrativa é a mesma, a seguir o processo com a mais baixa métrica instala a rota. Se a métrica é a mesma também, a seguir o processo de EIGRP com o mais baixo processo de EIGRP ID instala a rota no RIB. A tabela de topologia do outro processo de EIGRP terá a rota instalada com sucessores zero e um valor infinito FD.

Aqui está um exemplo:



```
R1#show ip eigrp topology 192.168.1.0 255.255.255.0
IP-EIGRP (AS 1): Topology entry for 192.16.1.0/24
State is Passive, Query origin flag is 1, 1 Successor(s), FD is 2681856
Routing Descriptor Blocks:
10.3.1.6 (Serial2/0), from 10.3.1.6, Send flag is 0x0
  Composite metric is (2681856/2169856), Route is Internal
  Vector metric:
    Minimum bandwidth is 1544 Kbit
    Total delay is 40000 microseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 1
IP-EIGRP (AS 2): Topology entry for 192.16.1.0/24
State is Passive, Query origin flag is 1, 0 Successor(s), FD is 4294967295
Routing Descriptor Blocks:
10.4.1.5 (Ethernet1/0), from 10.4.1.5, Send flag is 0x0
  Composite metric is (2681856/2169856), Route is Internal
  Vector metric:
    Minimum bandwidth is 1544 Kbit
```

Total delay is 40000 microseconds  
Reliability is 255/255  
Load is 1/255  
Minimum MTU is 1500  
Hop count is 1R1#**show ip route 192.168.1.0 255.255.255.0**

Routing entry for 192.168.1.0/24

Known via "eigrp 1", distance 90, metric 2681856, type internal

Redistributing via eigrp 1

Last update from 10.3.1.6 on Serial2/0, 00:04:16 ago

Routing Descriptor Blocks:

\* 10.3.1.6, from 10.3.1.6, 00:04:16 ago, via Serial2/0

Route metric is 2681856, traffic share count is 1

Total delay is 40000 microseconds, minimum bandwidth is 1544 Kbit

Reliability 255/255, minimum MTU 1500 bytes

Loading 1/255, Hops 1