

Compreendendo a CPU de VIP que executa em 99% e coloca em buffer no lado Rx

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Informações de Apoio](#)

[Fundamentos de arquitetura do Cisco 7500 Series](#)

[Tipos de buffers de pacotes](#)

[Corridas VIP na utilização CPU de 99%](#)

[Exemplos dos buffers no lado de Rx](#)

[Outras causas da utilização elevada da CPU em VIPs](#)

[Informações a serem coletadas se você abrir um pedido de serviço de TAC](#)

[Informações Relacionadas](#)

Introdução

Este documento explica porque o Versatile Interface Processor (VIP) CPU é executado em 99%, e o que são buffers do lado de Rx.

Pré-requisitos

Requisitos

Não existem requisitos específicos para este documento.

Componentes Utilizados

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

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

Convenções

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

Informações de Apoio

Élado RX em buffer o processo que ocorre quando a interface externa:

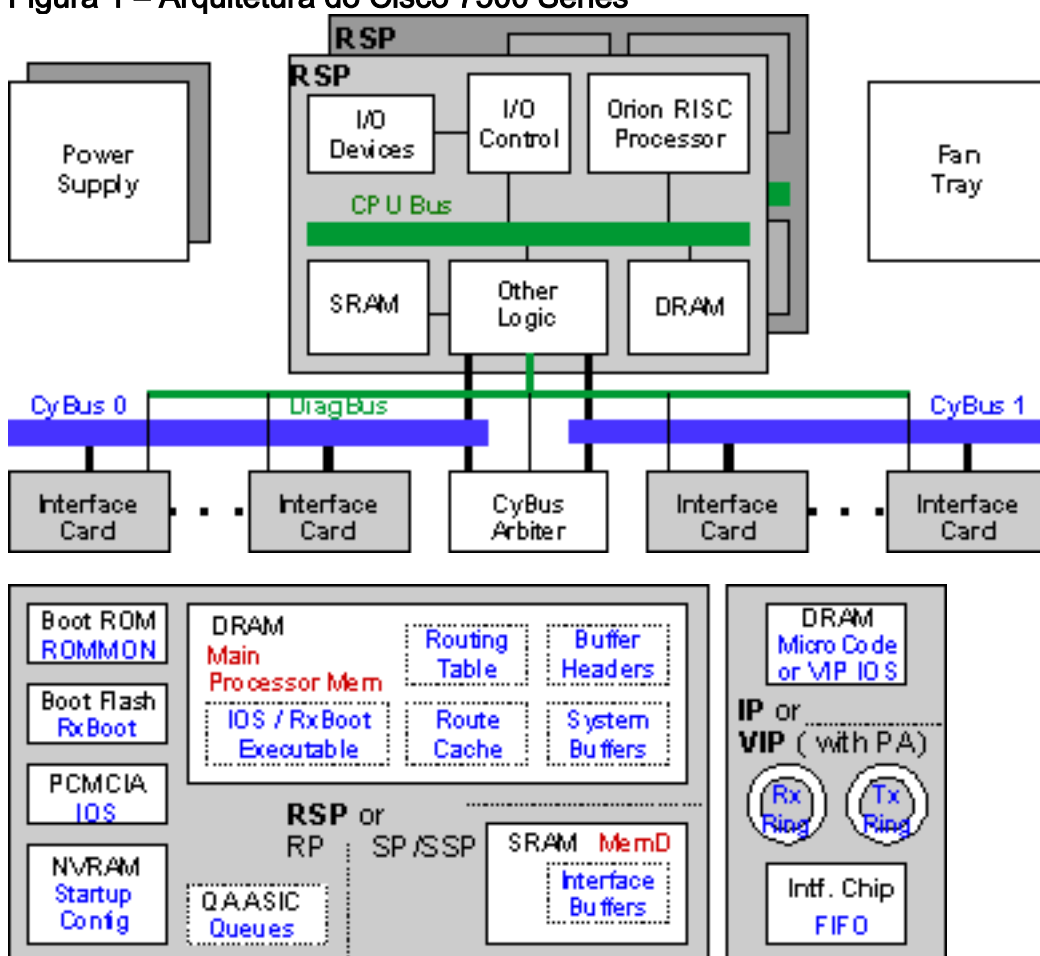
- é congestionado.
- usa o primeiro dentro, primeiramente para fora a estratégia de fila (FIFO).

O Versatile Interface Processor de entrada (VIP) não deixa cair o pacote imediatamente. Em lugar de, protege o pacote em sua memória de pacotes até que os buffers estejam disponíveis para a interface enviada. Baseado no tipo de VIP, a memória de pacotes pode ser ram estática (SRAM) ou ram dinâmica síncrona (SDRAM).

Fundamentos de arquitetura do Cisco 7500 Series

Cada processador de interface (IP ou VIP do legado) tem uma conexão a um barramento de sistema prolongado de alta velocidade chamado uns CyBus. A rota/processadores de switch (RSP) é conectada a dois CyBuses (veja [figura 1](#)).

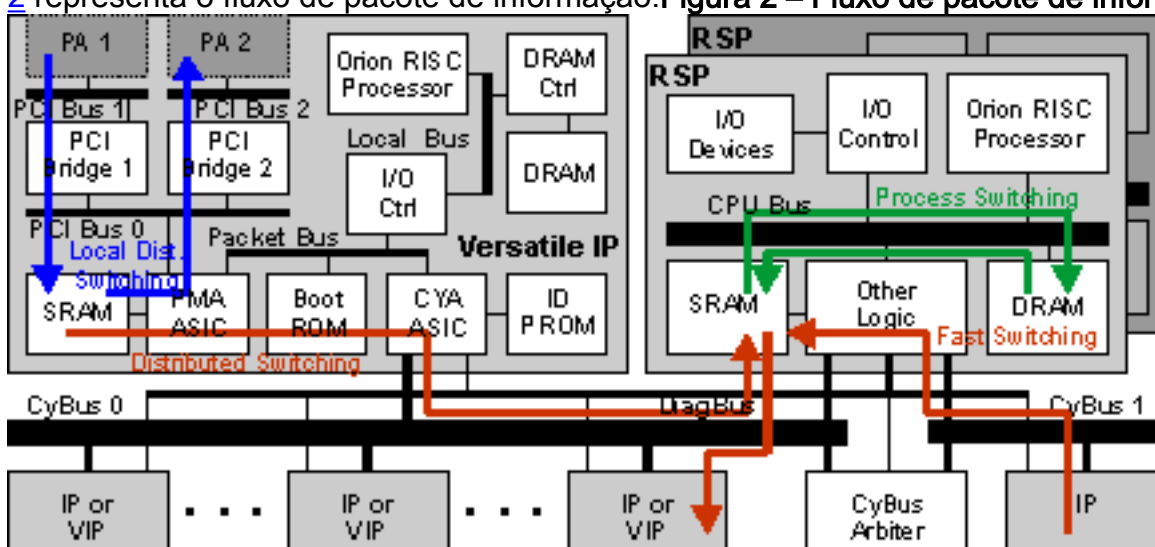
Figura 1 – Arquitetura do Cisco 7500 Series



Tipos de buffers de pacotes

Esta seção descreve os vários tipos de buffers de pacotes.

- **Bufferes de sistema na memória de processador no RSP** Estes bufferes são usados para pacotes comutados por processamento. Você pode ver estes bufferes na saída das **relações da mostra** (filas de entrada e de saída) e de [comandos show buffers](#). Um Cisco 7500 Series Router não deve fazer muito switching por processo. Consequentemente, se você tem problemas com bufferes de sistema, significa que pacotes demais estão enviados ao nível de processo. Isto pode ser devido a muitos fatores, como: Uma tempestade de broadcast Instabilidade na rede que causa atualizações de roteamento um ataque de "negação de serviço" (DoS) Uma característica que não seja apoiada no caminho de switching rápido (por exemplo, X.25) Pacotes IP com opções. Para obter informações sobre de como pesquisar defeitos o processo excessivo que comuta, refira estes documentos: [Troubleshooting de Alta Utilização de CPU em Cisco Routers](#) [Troubleshooting de Quedas de Fila de Entrada e Quedas de Fila de Saída](#)
- **Memória de pacotes em bufferes RSP (MEMD)** O tamanho de memd é fixo no 2 MB no RSP7000, no RSP1, no RSP2, e no RSP4. No RSP8 e RSP16, o tamanho do MEMD é 8 MB. O MEMD está distribuído entre todas as relações na altura da inicialização, quando há um Online Insertion and Removal (OIR), um reload do microcódigo, uma mudança da unidade de transmissão máxima (MTU), ou um complexo CBUS. Para obter mais informações sobre do complexo CBUS, refira [o que causa um "%RSP-3-RESTART: cbus complex"?](#). Você pode usar o [comando show controllers cbus](#) verificar o estado dos bufferes de memd. Quando o MEMD é atribuído, estas estruturas estão criadas: Um **local free queue (lfreq)** — É atribuído a cada relação, e usado para os pacotes recebidos nesta relação. Um **global free queue (gfreeq)** — É atribuído igualmente, e uma relação pode cair de volta a essa fila dentro de alguns limites. Um **transmitir fila (txqueue ou txq)** — é atribuída a cada relação, e usada para os pacotes que saem através desta relação. O **transmit accumulator (txacc)** — Representa o número de elementos no transmitir fila da interface de saída (txqueue). Quando o transmit accumulator (txacc) igualar o transmitir limit (txlimit), todos os bufferes estão livrados. Quando o txacc é 0, a fila está completa, e enfileirando-se é reservado.
- **Memória de pacotes** Em um VIP, a memória de pacotes contém os buffers de pacotes (partículas) usados para os pacotes recebidos de ou enviados a uma interface de VIP. [Figura 2](#) representa o fluxo de pacote de informação. **Figura 2 – Fluxo de pacote de informação**



Esta seção focaliza em um VIP onde o Distributed Switching seja permitido, porque lado RX em buffer ocorre geralmente quando um pacote segue este tipo de trajeto de switching. As encenações diferentes são possíveis, que são explicadas aqui: Cenário 1: Quando não houver nenhuma congestão na interface externa. Um pacote é recebido em um adaptador de porta (PA) e movido em um buffers de pacotes na memória de pacotes. Se o VIP não pode distribuir-

interruptor o pacote, ele para a frente o pacote ao RSP, que faz a decisão de switching. Se o VIP puder tomar a decisão da switching e a interface de saída estiver no mesmo VIP, o pacote é enviado para a interface de saída. O pacote seria "ligado localmente" o VIP, porque não cruza o cbus. Se o VIP puder tomar a decisão de switching e a interface de saída estiver em outro slot, o VIP tenta copiar o pacote sobre o cbus no txqueue (no MEMD) da interface de saída. O pacote então é copiado (V) ao IP que parte sobre o cbus e mandado na linha. Cenário 2: Quando a interface externa for congestionada. Há duas possibilidades: Se o enfileiramento estiver configurado na interface de saída, o VIP transfere o pacote no txqueue em MEMD e o pacote é puxado imediatamente da fila pelo código de enfileiramento. Se o enfileiramento com base em RSP é configurado, o pacote está copiado nos buffers de sistema na memória de processador no RSP. Se for utilizado enfileiramento baseado em VIP, o pacote é copiado para a memória de pacotes do VIP de saída. Se a estratégia de fila da interface externa é FIFO, a relação não deixa cair o pacote imediatamente (este é o que acontece normalmente com FIFO quando uma interface externa é congestionada). Em lugar de, o VIP de entrada protege o pacote em sua memória de pacotes até que alguns buffers estejam outra vez disponíveis para a interface enviada. Isto é chamado lado RX em buffer.

Use o **comando show controllers vip accumulator** verificar o estado de lado RX em buffer. O estado indica:

- O número de interfaces de saída apresenta no roteador.
- Quantos pacotes o VIP tem a colocação em buffer rx para estas relações.
- Porque o VIP tem a colocação em buffer rx.
- Quantos pacotes o VIP desligou e porquê.

Corridas VIP na utilização CPU de 99%

Uma consequência do buffer no lado Rx é que o VIP é executado com utilização de 99% da CPU. O VIP monitora continuamente o estado do txqueue da interface externa e, assim que houver um buffer livre, copia o pacote sobre o cbus no txqueue.

Não há nada que alarma-se em si mesmo quando o VIP é executado em 99% quando a colocação em buffer RX ocorre. Não significa que o VIP está sobrecarregado. Se o VIP recebe algo mais importante fazer (por exemplo, um outro pacote a comutar), este não está afetado pela alta utilização da CPU.

Está aqui um teste simples que você pode fazer no laboratório para ilustrar este:

A série 2/0/0 tem um Clock Rate dos kbps 128, e recebe o tráfego na linha taxa. O tráfego é comutado à série 10/0 onde o Clock Rate é 64 kbps, e a estratégia de fila é FIFO. A única opção é descartar os pacotes.

```
router#show controller cbus
```

```
MEMD at 40000000, 8388608 bytes (unused 697376, recarves 6, lost 0)
```

```
RawQ 48000100, ReturnQ 48000108, EventQ 48000110
```

```
BufhdrQ 48000130 (21 items), LovltrQ 48000148 (15 items, 2016 bytes)
```

```
IpcbufQ 48000158 (24 items, 4096 bytes)
```

IpcbufQ_classic 48000150 (8 items, 4096 bytes)
3570 buffer headers (48002000 - 4800FF10)
pool0: 8 buffers, 256 bytes, queue 48000138
pool1: 2940 buffers, 1536 bytes, queue 48000140
pool2: 550 buffers, 4512 bytes, queue 48000160
pool3: 4 buffers, 4544 bytes, queue 48000168
slot2: VIP2, hw 2.11, sw 22.20, ccb 5800FF40, cmdq 48000090, vps 8192
software loaded from system
IOS (tm) VIP Software (SVIP-DW-M), Version 12.0(21)S, EARLY DEPLOYMENT RELEASE
SOFTWARE (fc1)
ROM Monitor version 122.0
Mx Serial(4), HW Revision 0x3, FW Revision 1.45
Serial2/0/0, applique is V.35 DCE
received clockrate 2015232
gfreeq 48000140, lfreeq 480001D0 (1536 bytes)
rxlo 4, rxhi 336, rxcurr 16, maxrxcurr 293
txq 48001A00, txacc 48001A02 (value 294), txlimit 294
Serial2/0/1, applique is V.35 DTE
received clockrate 246
gfreeq 48000140, lfreeq 480001D8 (1536 bytes)
rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0
txq 48001A08, txacc 48001A0A (value 6), txlimit 6
Serial2/0/2, applique is Universal (cable unattached)
received clockrate 246
gfreeq 48000140, lfreeq 480001E0 (1536 bytes)
rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0
txq 48001A10, txacc 48001A12 (value 6), txlimit 6
Serial2/0/3, applique is Universal (cable unattached)
received clockrate 246
gfreeq 48000140, lfreeq 480001E8 (1536 bytes)
rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0
txq 48001A18, txacc 48001A1A (value 6), txlimit 6
slot10: FSIP, hw 1.12, sw 20.09, ccb 5800FFC0, cmdq 480000D0, vps 8192

software loaded from system

Serial10/0, applique is V.35 DTE

gfreeq 48000140, lfreeq 48000208 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 1, maxrxcurr 1

txq 48000210, txacc 480000B2 (value 2), txlimit 294

Serial10/1, applique is Universal (cable unattached)

gfreeq 48000140, lfreeq 48000218 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0

txq 48000220, txacc 480000BA (value 6), txlimit 6

Serial10/2, applique is Universal (cable unattached)

gfreeq 48000140, lfreeq 48000228 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0

txq 48000230, txacc 480000C2 (value 6), txlimit 6

Serial10/3, applique is Universal (cable unattached)

gfreeq 48000140, lfreeq 48000238 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0

txq 48000240, txacc 480000CA (value 6), txlimit 6

Serial10/4, applique is Universal (cable unattached)

gfreeq 48000140, lfreeq 48000248 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0

txq 48000250, txacc 480000D2 (value 6), txlimit 6

Serial10/5, applique is Universal (cable unattached)

gfreeq 48000140, lfreeq 48000258 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0

txq 48000260, txacc 480000DA (value 6), txlimit 6

Serial10/6, applique is Universal (cable unattached)

gfreeq 48000140, lfreeq 48000268 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0

txq 48000270, txacc 480000E2 (value 6), txlimit 6

Serial10/7, applique is Universal (cable unattached)

gfreeq 48000140, lfreeq 48000278 (1536 bytes)

rxlo 4, rxhi 336, rxcurr 0, maxrxcurr 0

```
txq 48000280, txacc 480000EA (value 6), txlimit 6
```

```
router#
```

O valor 2 significa que somente dois buffers estão deixados. A colocação em buffer RX não enfileira pacotes no MEMD quando o txacc é menos de 4.

O comando show controllers vip 2 tech-support do VIP mostra que é executado em 99% CPU:

```
router#show controllers vip 2 tech-support
```

```
show tech-support from Slot 2:
```

```
----- show version -----
```

```
Cisco Internetwork Operating System Software
```

```
IOS (tm) VIP Software (SVIP-DW-M), Version 12.0(21)S, EARLY DEPLOYMENT RELEASE  
SOFTWARE (fc1)
```

```
Copyright (c) 1986-2000 by cisco Systems, Inc.
```

```
Compiled Tue 18-Jul-00 22:03 by htseng
```

```
Image text-base: 0x600108F0, data-base: 0x602E0000
```

```
ROM: System Bootstrap, Version 11.1(4934) [pgreenfi 122], INTERIM SOFTWARE
```

```
VIP-Slot2 uptime is 1 week, 23 hours, 27 minutes
```

```
System returned to ROM by power-on
```

```
Running default software
```

```
cisco VIP2 (R4700) processor (revision 0x02) with 32768K bytes of memory.
```

```
Processor board ID 00000000
```

```
R4700 CPU at 100Mhz, Implementation 33, Rev 1.0, 512KB L2 Cache
```

```
4 Serial network interface(s)
```

```
Configuration register is 0x0
```

```
...
```

```
----- show process cpu -----
```

```
CPU utilization for five seconds: 99%/97%; one minute: 70%; five minutes: 69%
```

O VIP é executado na utilização CPU de 99% mesmo que receba somente os kbps 128. Isto mostra que a utilização CPU não está ligada ao número de pacotes por segundo. Isto é porque um VIP2 pode comutar muito mais pacotes do que este. É simplesmente um sinal de lado RX em buffer.

A fim verificar o que faz lado RX em buffer, para executar estes comandos:

```
router#show controllers vip 2 accumulator
```

```
show vip accumulator from Slot 2:
```

```
Buffered RX packets by accumulator:
```

```
...
Serial10/0:
MEMD txacc 0x00B2: 544980 in, 2644182 drops (126 paks, 378/376/376 bufs) 1544kbps
  No MEMD acc: 544980 in
    Limit drops : 2644102 normal pak drops, 80 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
  No MEMD buf: 0 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
...
Interface x:
MEMD txacc a: b in, c drops (d paks, e/f/g bufs) h kbps
No MEMD acc: i in
  Limit drops : j normal pak drops, k high prec pak drops
  Buffer drops : l normal pak drops, m high prec pak drops
No MEMD buf: n in
  Limit drops : o normal pak drops, p high prec pak drops
  Buffer drops : q normal pak drops, r high prec pak drops
```

C ha ve	Descrição
a	Endereço do txacc no MEMD. Há uma fila de buffers do lado da recepção de cada txacc do sistema (até 4096).
b	Número de pacotes que são colocação em buffer rx.
c	Número de pacotes que o VIP deixou cair. Se há bastante buffers de memória de pacote, o VIP pode colocação em buffer RX até o segundo do tráfego. Entretanto, se a interface estiver continuamente congestionada, não será possível evitar quedas.
d	Número dos pacotes de colocação em buffer rx atualmente.
e	Número de partículas atualmente no buffer de recepção. Um pacote pode ser constituído de partículas múltiplas.
f	Limite macio, que é o número máximo de partículas quando a memória de VIP for baixa.
g	Limite duro, que é o número máximo de partículas que podem ser usadas a qualquer hora.
h	A velocidade da interface de saída em kbps.
i	O número de pacotes da colocação em buffer rx porque nenhum txacc estava disponível no MEMD. Isto significa que a fila de saída esteve congestionada (não mais buffer livre está disponível no Tx-queue). A solução a este problema é aumentar a largura de banda da interface de saída (se possível).
j	O número de pacotes com a Precedência IP a não ser 6 ou 7 que não poderiam ser enviados porque

	não há nenhum CRNA MEMD, e é deixado cair porque o delicado ou o limite rígido de partículas foram alcançados.
k	Mesmos que j, mas para pacotes com Precedência IP 6 ou 7 (rede interna e rede).
l	O número de pacotes com a Precedência IP a não ser 6 ou 7 que o VIP quer à colocação em buffer RX, mas gotas devido a uma falta dos buffer livre na memória de pacotes. Dos Cisco IOS Software Releases 12.0(13)S e 12.1(4) avante, você pode igualmente usar o comando show controller vip [all/slot-] packet-memory-drops ver o número de pacotes deixados cair. Neste caso, uma atualização da memória do pacote ajuda.
m	Mesmos que l, mas para pacotes com Precedência IP 6 ou 7 (rede interna e rede).
n	O número de pacotes as tentativas VIP à colocação em buffer RX porque não há nenhum buffer de memd, mas não pode fazer tão devido aos bufferes de uma falta de pacote de memória. Promova a memória de pacotes neste caso. Dos Cisco IOS Software Releases 12.0(13)S e 12.1(4) avante, você pode igualmente usar o comando show controllers vip [all/slot-] packet-memory-drops compreender porque os pacotes foram deixados cair.
o	O número de pacotes da colocação em buffer rx com a Precedência IP a não ser 6 ou 7 sem o buffer de memd que são deixados cair porque o (f) macio ou (g) o limite duro foram alcançados. Nesta situação, um RSP16 ajuda porque tem mais memória MEMD (8 MB comparado ao 2 MB para o RSP1, o RSP2, o RSP4, e o RSP7000). Você pode igualmente reduzir o MTU de algumas relações (tais como o ATM, o POS, ou o FDDI) neste caso. Estas relações têm geralmente um 4470-byte MTU, e menos bufferes de memd podem ser atribuídos porque os bufferes têm que ser maiores.
p	Mesmos que o, mas para pacotes com Precedência IP 6 ou 7 (rede interna e rede).
q	O número de pacotes com a Precedência IP a não ser 6 ou 7 que o VIP tenta à colocação em buffer RX porque não há nenhum buffer de memd, mas não pode fazer tão devido aos bufferes de uma falta de pacote de memória. Uma elevação da memória de pacotes ajuda neste caso. Dos Cisco IOS Software Releases 12.0(13)S e 12.1(4) avante, você pode igualmente usar o comando show controllers vip [all/slot-] packet-memory-drops compreender melhor porque os pacotes foram deixados cair.
r	Mesmos que q, mas para pacotes com Precedência IP 6 ou 7 (rede interna e rede).

Se o roteador executa uma versão de Cisco IOS Software mais cedo do que o 12.0(13)ST, 12.1(04)DB, 12.1(04)DC, 12.0(13)S, 12.1(4) 12.1(4)AA 12.1(4)T 012.0(13), ou 12.0(13)SC, a saída do **acumulador do [all/slot-] vip dos controladores da mostra** fornecem uma versão simplificada do acima. Não leva em consideração os precedentes diferentes IP dos pacotes descartado devido a lado RX em buffer.

A saída se parece com isto:

```
router#show controllers vip 2 accumulator

show vip accumulator from Slot 2:

Buffered RX packets by accumulator:
...
Serial10/0:
MEMD txacc 0x00B2: 544980 in, 2644182 drops (126 paks, 378/376/376 bufs) 1544kbps
  No MEMD acc: 544980 in
    Limit drops : 2644102 normal pak drops, 80 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
  No MEMD buf: 0 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
...
Interface x:
MEMD txacc a: b in, c drops (d paks, e/f/g bufs) h kbps
No MEMD acc: i in
  Limit drops : j normal pak drops, k high prec pak drops
  Buffer drops : l normal pak drops, m high prec pak drops
No MEMD buf: n in
  Limit drops : o normal pak drops, p high prec pak drops
  Buffer drops : q normal pak drops, r high prec pak drops
```

Exemplos dos bufferes no lado de Rx

Exemplo 1: O VIP no entalhe 2 recebe o tráfego em um 128Kbps e distribui-o à série 10/0 (64Kbps).

```
router#show controllers vip 2 accumulator

show vip accumulator from Slot 2:

Buffered RX packets by accumulator:
...
Serial10/0:
MEMD txacc 0x00B2: 544980 in, 2644182 drops (126 paks, 378/376/376 bufs) 1544kbps
  No MEMD acc: 544980 in
    Limit drops : 2644102 normal pak drops, 80 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
  No MEMD buf: 0 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
...
Interface x:
MEMD txacc a: b in, c drops (d paks, e/f/g bufs) h kbps
No MEMD acc: i in
  Limit drops : j normal pak drops, k high prec pak drops
  Buffer drops : l normal pak drops, m high prec pak drops
No MEMD buf: n in
  Limit drops : o normal pak drops, p high prec pak drops
  Buffer drops : q normal pak drops, r high prec pak drops
```

- Aqui, 544980 pacotes são com sucesso colocação em buffer rx e 2644182 são deixados cair. 80 pacotes fora dos 2644182 que são deixados cair tiveram uma Precedência IP de 6 ou de 7.
- 126 pacotes são atualmente colocação em buffer rx e usam 378 partículas.
- Todos os pacotes são colocação em buffer rx devido a uma falta do buffer livre no Tx-queue no MEMD. Isso significa que a interface de saída está congestionada. As gotas ocorrem porque o número máximo de pacotes da colocação em buffer rx é alcançado. A solução típica é promover a largura de banda da interface externa, redistribui algum tráfego de modo que a interface externa seja congestionada menos, ou permitir algum que enfileira-se para deixar cair o tráfego menos importante.

Exemplo 2: Bufferes do lado de Rx sem gotas.

```
router#show controllers vip 2 accumulator
```

```
show vip accumulator from Slot 2:
```

```
Buffered RX packets by accumulator:
```

```
...
Serial10/0:
MEMD txacc 0x00B2: 544980 in, 2644182 drops (126 paks, 378/376/376 bufs) 1544kbps
  No MEMD acc: 544980 in
    Limit drops : 2644102 normal pak drops, 80 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
  No MEMD buf: 0 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
...
Interface x:
MEMD txacc a: b in, c drops (d paks, e/f/g bufs) h kbps
No MEMD acc: i in
  Limit drops : j normal pak drops, k high prec pak drops
  Buffer drops : l normal pak drops, m high prec pak drops
No MEMD buf: n in
  Limit drops : o normal pak drops, p high prec pak drops
  Buffer drops : q normal pak drops, r high prec pak drops
```

- Neste exemplo, 85709 pacotes são colocação em buffer rx porque o ATM1/0 é congestionado mas nenhum pacote é deixado cair.
- 117795 pacotes são colocação em buffer rx porque o VIP não pode obter um buffer de memd. Nenhum pacote é deixado cair. Uma solução típica é diminuir alguns dos MTU de modo que mais bufferes de memd possam ser atribuídos. Igualmente ajudas Um RSP8.

Exemplo 3: Switching local.

```
router#show controllers vip 2 accumulator
```

```
show vip accumulator from Slot 2:
```

```
Buffered RX packets by accumulator:
```

```
...
Serial10/0:
MEMD txacc 0x00B2: 544980 in, 2644182 drops (126 paks, 378/376/376 bufs) 1544kbps
  No MEMD acc: 544980 in
    Limit drops : 2644102 normal pak drops, 80 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
  No MEMD buf: 0 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
...
```

```

Interface x:
MEMD txacc a: b in, c drops (d paks, e/f/g bufs) h kbps
No MEMD acc: i in
    Limit drops : j normal pak drops, k high prec pak drops
    Buffer drops : l normal pak drops, m high prec pak drops
No MEMD buf: n in
    Limit drops : o normal pak drops, p high prec pak drops
    Buffer drops : q normal pak drops, r high prec pak drops

```

O txacc local significa que esta interface de saída está no mesmo VIP que a relação onde os pacotes são recebidos. Estes pacotes são comutados localmente, mas a interface externa (neste caso, srp 0/0/0) é congestionada. 2529 pacotes são colocação em buffer rx, e nenhum pacote é deixado cair.

Exemplo 4: Filas dianteiras.

```

router#show controllers vip 2 accumulator
Buffered RX packets by accumulator:
Forward queue 0 : 142041 in, 3 drops (0 paks, 0/24414/24414 bufs) 100000kbps
  No MEMD buf: 142041 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 3 normal pak drops, 0 high prec pak drops
Forward queue 9 : 68 in, 0 drops (0 paks, 0/15/484 bufs) 1984kbps
  No MEMD buf: 68 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
Forward queue 13: 414 in, 0 drops (0 paks, 0/14/468 bufs) 1920kbps
  No MEMD buf: 414 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops
Forward queue 14: 46 in, 0 drops (0 paks, 0/14/468 bufs) 1920kbps
  No MEMD buf: 46 in
    Limit drops : 0 normal pak drops, 0 high prec pak drops
    Buffer drops : 0 normal pak drops, 0 high prec pak drops

```

Alguns pacotes não podem ser comutados por distribuição. Neste caso, o VIP tem que enviar os pacotes à fila bruta do RSP, que faz então a decisão de switching. Quando os pacotes não podem ser copiados imediatamente no MEMD, os RX-buffers VIP eles e mantêm-se a par do quantos pacotes são colocação em buffer rx pela interface de entrada.

As filas de encaminhamento de 0 a 7 são para o primeiro adaptador de porta (PA) e as de 8 a 15 para o segundo PA.

Número de fila de encaminhamento	... mostra o número de pacotes da colocação em buffer rx que são recebidos no...
0	primeiro orifício de plugue do primeiro PA (Adaptador de porta)
8	primeira abertura bloqueada do segundo PA
9	segunda abertura do segundo PA

Outras causas da utilização elevada da CPU em VIPs

Quando é encontrado lado RX em buffer para ser inativo, um destes fatores pode causar a utilização elevada da CPU no VIP:

- **Utilização CPU de 99% no VIP, causado pelo Distributed Traffic Shaping** Quando o Distributed Traffic Shaping (DTS) é configurado, o CPU VIP salta a 99% assim que um pacote entrar na fila do dTS. Este é o correto e o comportamento esperado. Quando o dTS é configurado, o CPU VIP gerencia para verificar se a próxima vez que o intervalo (Tc) chega quando o CPU não é ocupado (isto é, quando há um sem tráfego). Se não, a verificação é rebocada nas rotinas da interrupção do Tx/Rx. Você gerencie o CPU somente quando não é ocupado. Conseqüentemente, o desempenho não é afetado. Para compreender o que “o intervalo de tempo seguinte” significa, veja [o que é um Token Bucket?](#) **Nota:** O modelagem de tráfego torna-se ativo somente quando tem que enviar à fila um pacote na fila moldada. Ou seja quando a quantidade de tráfego exceder a taxa moldada. Isto explica porque o CPU VIP não é sempre 99% quando o dTS é configurado. Para obter mais informações sobre do dTS, veja: [Molde de tráfego distribuído Configurando o Distributed Traffic Shaping](#)
- **Utilização elevada da CPU no VIP causado por acessos de memória artificiais e por erros de alinhamento** Os erros de alinhamento e os acessos de memória artificiais falham as falhas de software que são corrigidas pelo Cisco IOS Software sem a necessidade de causar um crash o VIP. Se estes erros aparecem frequentemente, faz com que o sistema operacional faça muitas correções que podem conduzir à utilização elevada da CPU. Para obter mais informações sobre dos erros de alinhamento e dos acessos de memória artificiais, veja [acessos artificiais, erros de alinhamento, e interrupções espúria do Troubleshooting](#). Para verificar para ver se há acessos de memória artificiais e erros de alinhamento, use o **comando show alignment**. Um exemplo de tal erro olha como este:
`VIP-Slot1#show alignment`
No alignment data has been recorded.
No spurious memory references have been recorded.

As outras causas da utilização elevada da CPU podem ser a quantidade e a extensão das características distribuídas que são permitidas. Se você suspeita que esta poderia ser a razão, ou se você não pode identificar algumas das causas para a utilização elevada da CPU explicadas neste documento, abra um pedido do serviço com o centro de assistência técnica da Cisco (TAC).

[Informações a serem coletadas se você abrir um pedido de serviço de TAC](#)

Se você ainda precisa o auxílio depois que você segue os passos de Troubleshooting acima e os quer [abrir um pedido do serviço \(clientes registrados somente\)](#) com o tac Cisco, seja certo incluir esta informação:

- Saída do comando `show controllers vip [all/slot-] accumulator`
- Saída do comando `show technical-support` do RSP e do VIP relevantes

Anexe os dados coletados à sua requisição de serviço em um texto não compactado e simples (.txt). A fim anexar a informação a seu pedido do serviço, transfira-o arquivos pela rede através da [ferramenta do pedido do serviço TAC \(clientes registrados somente\)](#). Se você não pode alcançar a ferramenta do pedido do serviço, você pode anexar a informação relevante a seu pedido do serviço, e envia-o a attach@cisco.com com seu número do pedido do serviço na linha de assunto de sua mensagem.

Nota: Por favor não recarregue manualmente ou ciclo de energia o roteador antes que você recolha a informação acima (a menos que exigido para restaurar a operação de rede), porque esta pode fazer com que a informação importante esteja perdida que é precisada de determinar a causa de raiz do problema.

[Informações Relacionadas](#)

- [Sustentação do produto dos roteadores Cisco](#)
- [Pesquise defeitos e alertas: Cisco 7500 Series Routers](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)