

Troubleshooting de Falhas de Manutenção de Atividade Punt no Cisco IOS XE

Contents

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Arquivo de log de depuração Punt](#)

[A LSMPI \(Linux Shared Memory Punt Interface, interface de punt de memória compartilhada\)](#)

[O Vigilante de Punt](#)

[Embedded Event Manager \(EEM\) para Coleta de Dados](#)

[Um exemplo prático](#)

[Melhorias](#)

Introdução

Este documento descreve como solucionar problemas de falhas de punt keep alive.

Pré-requisitos

Requisitos

Conhecimento básico no Cisco IOS® XE.

Componentes Utilizados

Este documento é baseado nos roteadores Cisco IOS XE como CSR8000v, ASR1000 e ISR4000 Series.

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 rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Informações de Apoio

O caminho de punt em sistemas baseados no Cisco IOS XE é um caminho de dados interno. Esse é o caminho onde ocorre a comunicação entre o plano de controle e o plano de dados.

Esse caminho interno é usado para transmitir pacotes de plano de controle para consumo do roteador.

Quando esse caminho falha, você pode ver esse tipo de erro no registro.

```
%IOSXE_INFRA-4-NO_PUNT_KEEPLIVE: Keepalive not received for 60 seconds
```

As mensagens keep alive são mensagens que monitoram a integridade do caminho entre o QFP e o RP.

Esse caminho é crítico para o sistema operar.

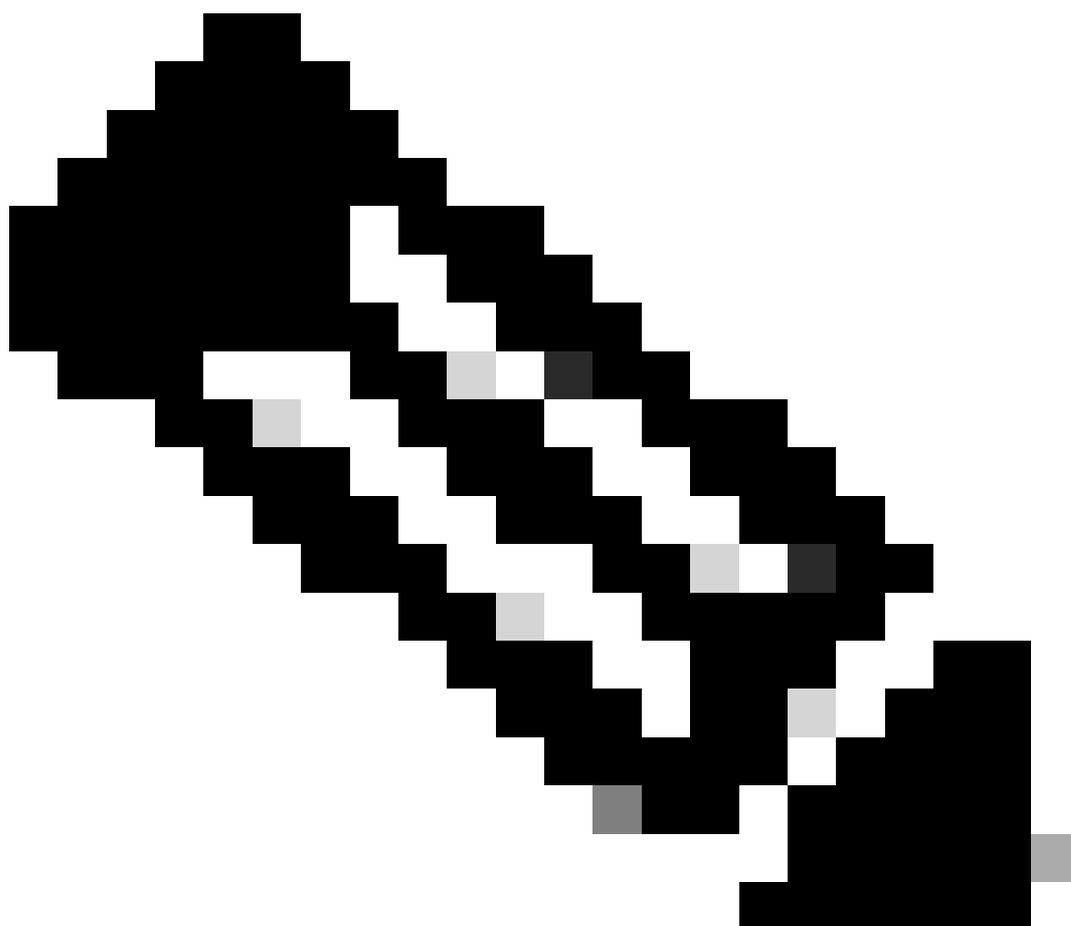
Se esses keep alives não forem recebidos em 5 minutos, você poderá ver um log crítico como este:

```
%IOSXE_INFRA-2-FATAL_NO_PUNT_KEEPLIVE: Keepalive not received for 300 seconds resetting
```

O sistema é redefinido para se recuperar dessa condição.

O arquivo de log de depuração Punt

No caso de falhas de manutenção de atividade de punt e reinicializações devido a elas, o sistema cria um arquivo chamado punt_debug.log que coleta dados relevantes para entender o comportamento no momento do problema.



Note: Certifique-se de ter o sistema atualizado com a versão mais recente do software Cisco IOS XE para que o arquivo `punt_debug.log` seja gerado.

Esse arquivo contém esses comandos executados várias vezes para entender contadores diferentes.

```
show platform software infra punt-keepalive
```

```
show platform software infra lsmpi
```

```
show platform software infrastructure lsmpi driver
```

```
show platform software infra lsmpi bufusage
```

```
show platform software punt-policer
```

```
show platform software status control-processor brief
```

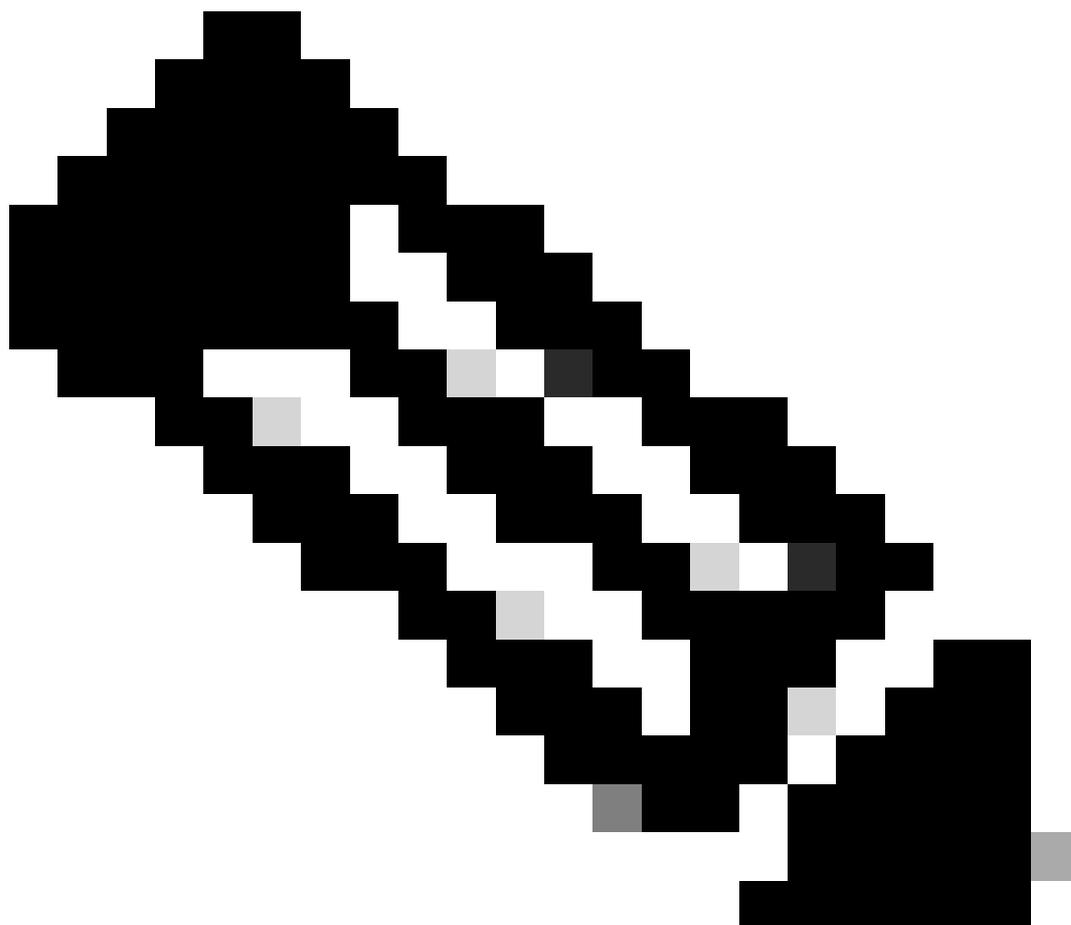
```
show process cpu platform sorted
```

```
show platform software infrastructure punt
```

```
show platform hardware qfp ative statistics drop
```

```
show platform hardware qfp ative infra punt statistics type per-cause
```

```
show platform hardware qfp ative infrastructure bqs queue output default all
```



Note: No arquivo `punt_debug.log`, você se concentra nos indicadores de erro e na grande quantidade de pacotes que podem causar o problema.

A LSMPI (Linux Shared Memory Punt Interface, interface de punt de memória compartilhada)

Este componente é usado para transmitir pacotes e mensagens do processador de encaminhamento para o processador de roteamento.

O Vigilante de Punt

O punt policer é um mecanismo de proteção do plano de controle que permite que o sistema proteja e policie os pacotes do plano de controle.

Com o comando `show platform software punt-policer`, você pode ver os pacotes em conformidade e os descartados devido a esse vigilante.

```
----- show platform software punt-policer -----
```

Per Punt-Cause Policer Configuration and Packet Counters

Punt Cause	Description	Config Rate(pps)		Conform Packets		Dropped Pack
		Normal	High	Normal	High	Normal
2	IPv4 Options	874	655	0	0	0
3	Layer2 control and legacy	8738	2185	0	0	0
4	PPP Control	437	1000	0	0	0

-- snip : output omitted for brevity --

O comando `show platform software infrastructure punt` mostra dados do contador sobre causas de punt.

```
----- show platform software infrastructure punt -----
```

LSMPI interface internal stats:

enabled=0, disabled=0, throttled=0, unthrottled=0, state is ready

Input Buffers = 51181083

Output Buffers = 51150283

-- snip : output omitted for brevity --

EPC CP RX Pkt cleansed 0

Punt cause out of range 0

IOSXE-RP Punt packet causes:

3504959 ARP request or response packets

27 Incomplete adjacency packets

-- snip : output omitted for brevity --

FOR_US Control IPv4 protocol stats:

2369262 TCP packets

FOR_US Control IPv6 protocol stats:

6057 ICMPV6 packets

Packet histogram(500 bytes/bin), avg size in 119, out 95:

Pak-Size	In-Count	Out-Count
0+:	51108211	51144723

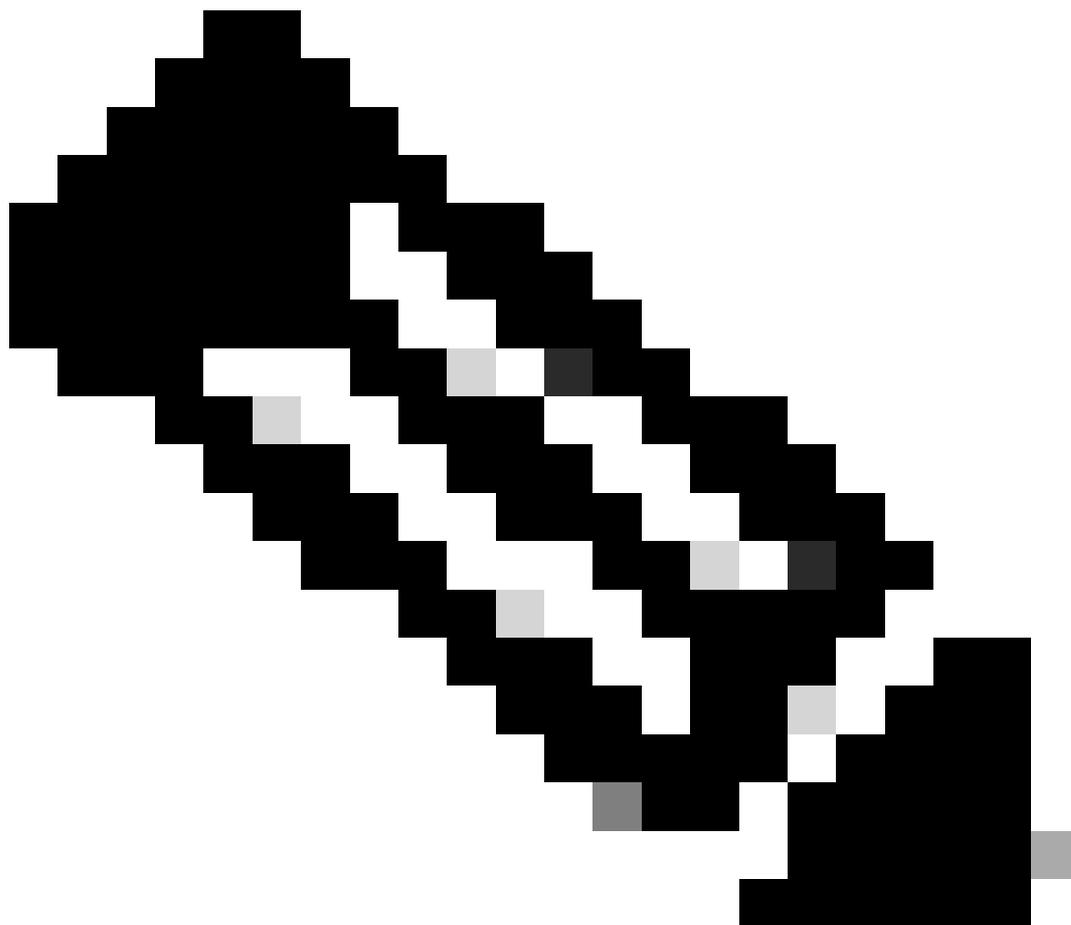
500+:	22069	2632
1000+:	2172	0
1500+:	3170	0

Esses dados são relevantes para entender o que pode estar afetando o caminho do punt keep alive.

Embedded Event Manager (EEM) para Coleta de Dados

Caso o arquivo punt_debug.log não forneça dados suficientes para diagnosticar o problema, o script EEM pode ser usado para obter mais pontos de dados no momento do problema.

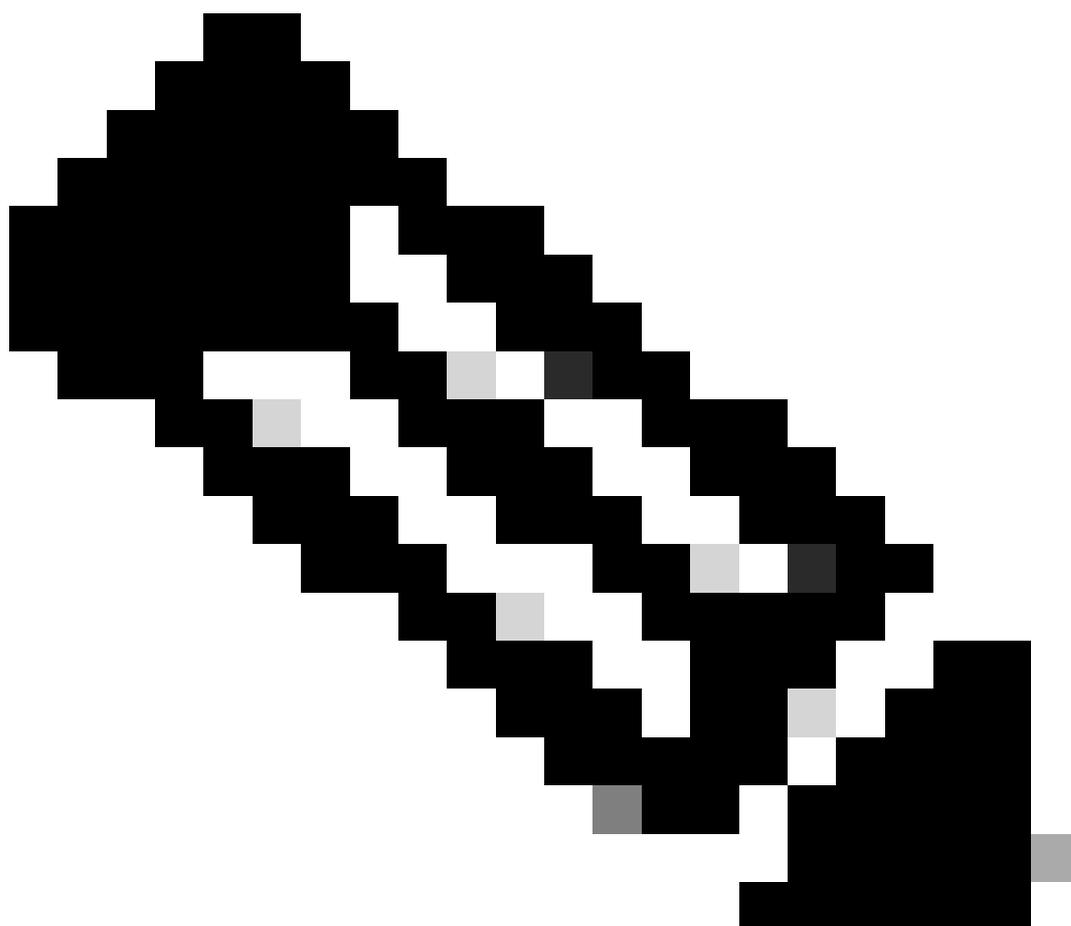
```
event manager applet punt_script authorization bypass
event syslog pattern "IOSXE_INFRA-4-NO_PUNT_KEEPLIVE" maxrun 1000
action 0.0 cli command "enable"
action 0.1 set i "0"
action 0.2 cli command "test platform software punt-keepalive ignore-fault"
action 0.3 while $i lt 10
action 0.4 syslog msg "iteration $i"
action 0.9 cli command "show clock | append bootflash:qfp_lsmpi.txt"
action 1.0 cli command "show platform software infrastructure lsmpi | append bootflash:qfp_lsmpi.txt"
action 1.1 cli command "show platform software infrastructure lsmpi driver | append bootflash:qfp_lsmpi.txt"
action 1.2 cli command "show platform software infrastructure lsmpi driver 0 | append bootflash:qfp_lsmpi.txt"
action 1.3 cli command "show platform software infrastructure lsmpi bufusage | append bootflash:qfp_lsmpi.txt"
action 1.4 cli command "show platform software infrastructure lsmpi bufusage 0 | append bootflash:qfp_lsmpi.txt"
action 1.5 cli command "show platform software infrastructure punt-keepalive | append bootflash:qfp_lsmpi.txt"
action 1.6 cli command "show platform software infrastructure punt | append bootflash:qfp_lsmpi.txt"
action 1.7 cli command "show platform software punt-policer | append bootflash:qfp_lsmpi.txt"
action 1.8 cli command "show platform hardware qfp active infrastructure punt stat type per-cause | append bootflash:qfp_lsmpi.txt"
action 1.9 cli command "show platform hardware qfp active infrastructure punt statistics type punt-drop | append bootflash:qfp_lsmpi.txt"
action 1.a cli command "show platform hardware qfp active infrastructure punt statistics type inject-drop | append bootflash:qfp_lsmpi.txt"
action 1.b cli command "show platform hardware qfp active infrastructure bqs queue output default interface | append bootflash:qfp_lsmpi.txt"
action 1.c cli command "show platform hardware qfp active statistics drop | append bootflash:qfp_lsmpi.txt"
action 1.d cli command "show platform hardware qfp active datapath utilization | append bootflash:qfp_lsmpi.txt"
action 1.e cli command "show platform hardware qfp active datapath infrastructure sw-hqf | append bootflash:qfp_lsmpi.txt"
action 1.f cli command "show platform hardware qfp active datapath infrastructure sw-distrib | append bootflash:qfp_lsmpi.txt"
action 1.g cli command "show platform hardware qfp active datapath infrastructure sw-pktmem | append bootflash:qfp_lsmpi.txt"
action 1.h cli command "show platform software status control-processor brief | append bootflash:qfp_lsmpi.txt"
action 2.0 increment i
action 2.1 wait 3
action 2.4 end
action 3.0 syslog msg "End of data collection. Please transfer the file at bootflash:qfp_lsmpi.txt"
action 5.0 cli command "debug platform hardware qfp active datapath crashdump"
```



Note: Os comandos contidos no script variam de acordo com a plataforma em que ele está configurado.

Esse script permite que você compreenda o lsmipi, os recursos e o estado de punt durante o tempo de ocorrência.

O script EEM inclui o comando `debug platform hardware qfp active datapath crashdump` que gera o dump central qfp, necessário para a equipe do desenvolvedor e o TAC.



Note: Se estiver preenchendo um caso com o TAC da Cisco, forneça o arquivo central gerado pelo script.

Se um rastreamento de pacote for necessário, esta emenda pode ser adicionada ao script:

Primeiro, defina a configuração de rastreamento de pacotes, que pode ser feita a partir do script EEM:

```
debug platform packet-trace packet 8192 fia-trace circular
debug platform condition both
debug platform packet-trace copy packet ambos L2
```

Em seguida, inicie e pare-o com estas ações no script EEM:

```
ação 6.2 comando cli "debug platform condition start"
ação 6.3 aguardar 8
ação 6.4 comando cli "debug platform condition stop"
```

Em seguida, descarte os dados com estes comandos em um arquivo separado:

```
action 6.5 cli command "show platform packet-trace statistics | append bootflash:traceAll.txt"  
ação 6.6 comando cli "show platform packet-trace summary | append bootflash:traceAll.txt"  
action 6.7 cli command "decodificação total show platform packet-trace packet | append  
bootflash:traceAll.txt"
```

Essa lógica de ações de rastreamento de pacote é adicionada logo após a instrução end do ciclo while no script EEM.

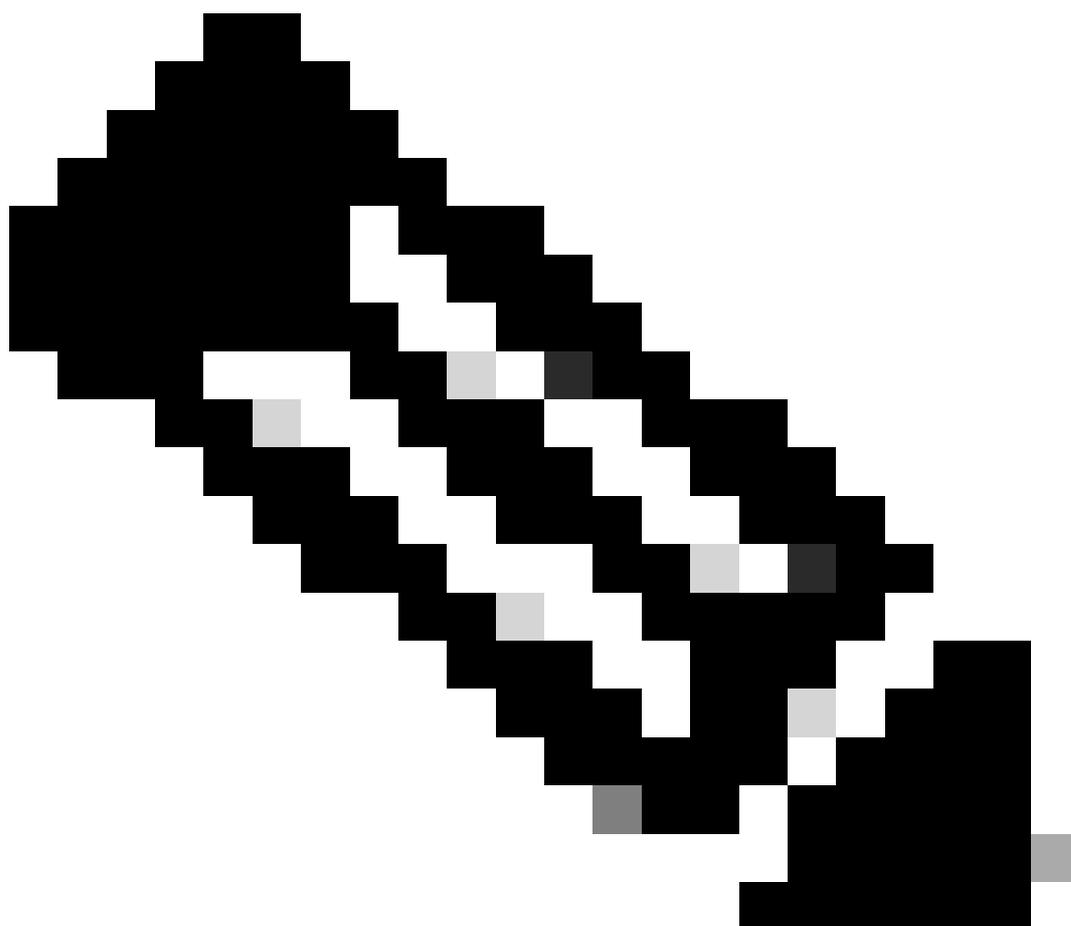
Esse script permite que você entenda que tipo de pacotes pode estar causando o problema.

O rastreamento de pacotes é um recurso documentado em [Troubleshooting com o IOS XE Datapath Packet Trace Feature](#)

Um exemplo prático

Um CSR8000v é reinicializado constantemente.

Depois de extrair o relatório do sistema, você pode observar um crashdump e um arquivo central iosd indicando funções relacionadas ao punt keep alive no rastreamento de pilha.



Note: Para a decodificação de rastreamento de pilha, é necessária a assistência do TAC.

No entanto, o arquivo crashinfo está em texto claro e você pode ver estes sintomas:

```
Jan 15 14:29:41.756 AWST: %IOSXE_INFRA-4-NO_PUNT_KEEPALIVE: Keepalive not received for 160 seconds
Jan 15 14:30:01.761 AWST: %IOSXE_INFRA-4-NO_PUNT_KEEPALIVE: Keepalive not received for 180 seconds
Jan 15 14:30:21.766 AWST: %IOSXE_INFRA-4-NO_PUNT_KEEPALIVE: Keepalive not received for 200 seconds
Jan 15 14:30:41.776 AWST: %IOSXE_INFRA-4-NO_PUNT_KEEPALIVE: Keepalive not received for 220 seconds
Jan 15 14:31:01.780 AWST: %IOSXE_INFRA-4-NO_PUNT_KEEPALIVE: Keepalive not received for 240 seconds
Jan 15 14:31:41.789 AWST: %IOSXE_INFRA-4-NO_PUNT_KEEPALIVE: Keepalive not received for 280 seconds
Jan 15 14:32:01.791 AWST: %IOSXE_INFRA-4-NO_PUNT_KEEPALIVE: Keepalive not received for 300 seconds
Jan 15 14:32:01.791 AWST: %IOSXE_INFRA-2-FATAL_NO_PUNT_KEEPALIVE: Keepalive not received for 300 seconds
```

%Software-forced reload

Exception to IOS Thread:

Frame pointer 0x7F0AE0EE29A8, PC = 0x7F0B342C16D2

UNIX-EXT-SIGNAL: Aborted(6), Process = PuntInject Keepalive Process

O aprimoramento para a geração automática de arquivos qfp core foi introduzido a partir da versão 17.15 do Cisco IOS XE através da ID de bug [CSCwf85505](#)

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.