

Dépannage des défaillances de maintien de connexion dans Cisco IOS XE

Table des matières

[Introduction](#)

[Conditions préalables](#)

[Exigences](#)

[Composants utilisés](#)

[Informations générales](#)

[Fichier journal de débogage Punt](#)

[Interface LSMPI \(Linux Shared Memory Punt Interface\)](#)

[Le contrôleur Punt](#)

[Gestionnaire d'événements intégré \(EEM\) pour la collecte de données](#)

[Un exemple pratique](#)

[Amélioration](#)

Introduction

Ce document décrit comment dépanner les pannes punt keep alive.

Conditions préalables

Exigences

Connaissances de base de Cisco IOS® XE.

Composants utilisés

Ce document est basé sur les routeurs Cisco IOS XE tels que les gammes CSR8000v, ASR1000 et ISR4000.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

Dans les systèmes basés sur Cisco IOS XE, le chemin ponctuel est un chemin de données interne. Il s'agit du chemin sur lequel s'effectue la communication entre le plan de contrôle et le plan de données.

Ce chemin interne est utilisé pour transmettre les paquets du plan de contrôle pour la consommation du routeur.

Lorsque ce chemin échoue, vous pouvez voir ce type d'erreur dans le journal.

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

Les messages keep alive sont des messages qui surveillent l'état du chemin entre le QFP et le RP.

Ce chemin est essentiel au fonctionnement du système.

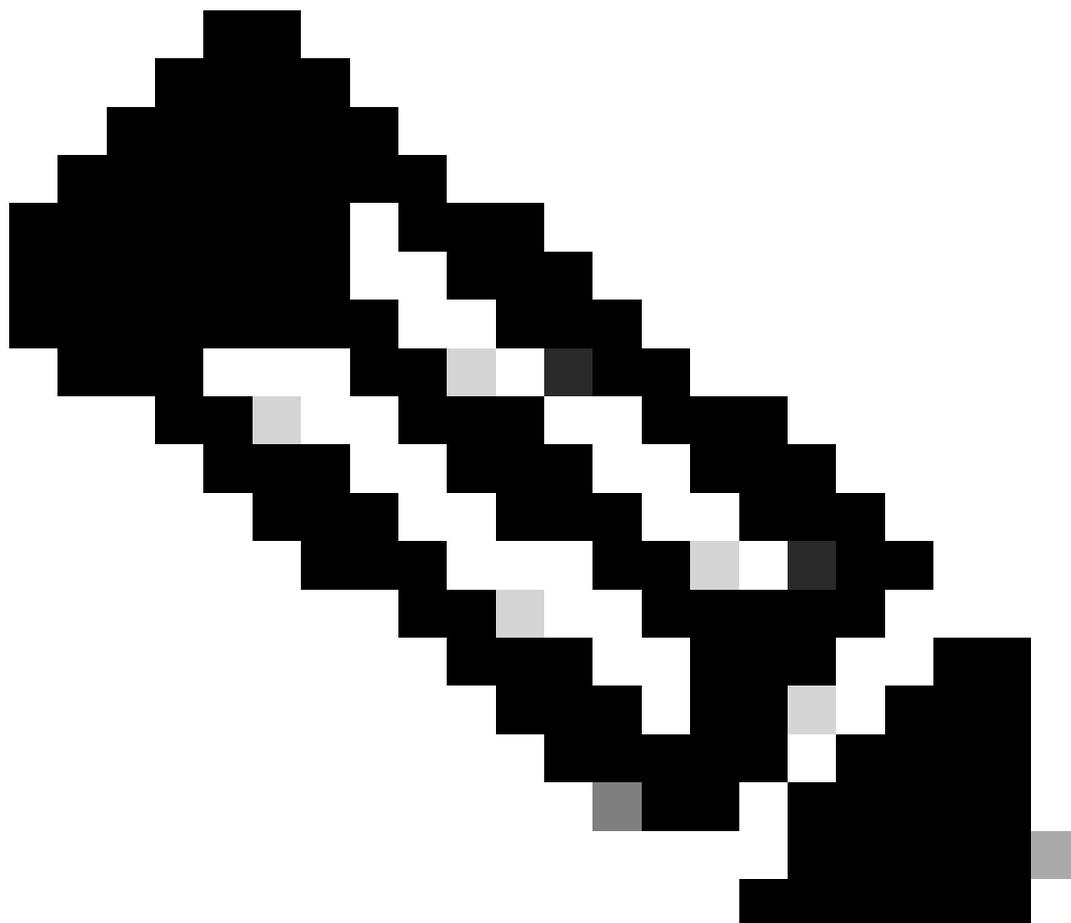
Si ces messages de test d'activité ne sont pas reçus dans les 5 minutes, vous pouvez voir un journal critique comme celui-ci :

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

Le système se réinitialise afin de récupérer de cette condition.

Le fichier journal de débogage Punt

En cas de panne punt keep alive et de réinitialisations dues à celle-ci, le système crée un fichier appelé punt_debug.log qui collecte les données pertinentes pour comprendre le comportement au moment du problème.



Remarque : Assurez-vous que le système est à jour avec la dernière version du logiciel Cisco IOS XE pour le fichier `punt_debug.log` à générer.

Ce fichier contient ces commandes exécutées plusieurs fois afin de comprendre différents compteurs.

```
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
```

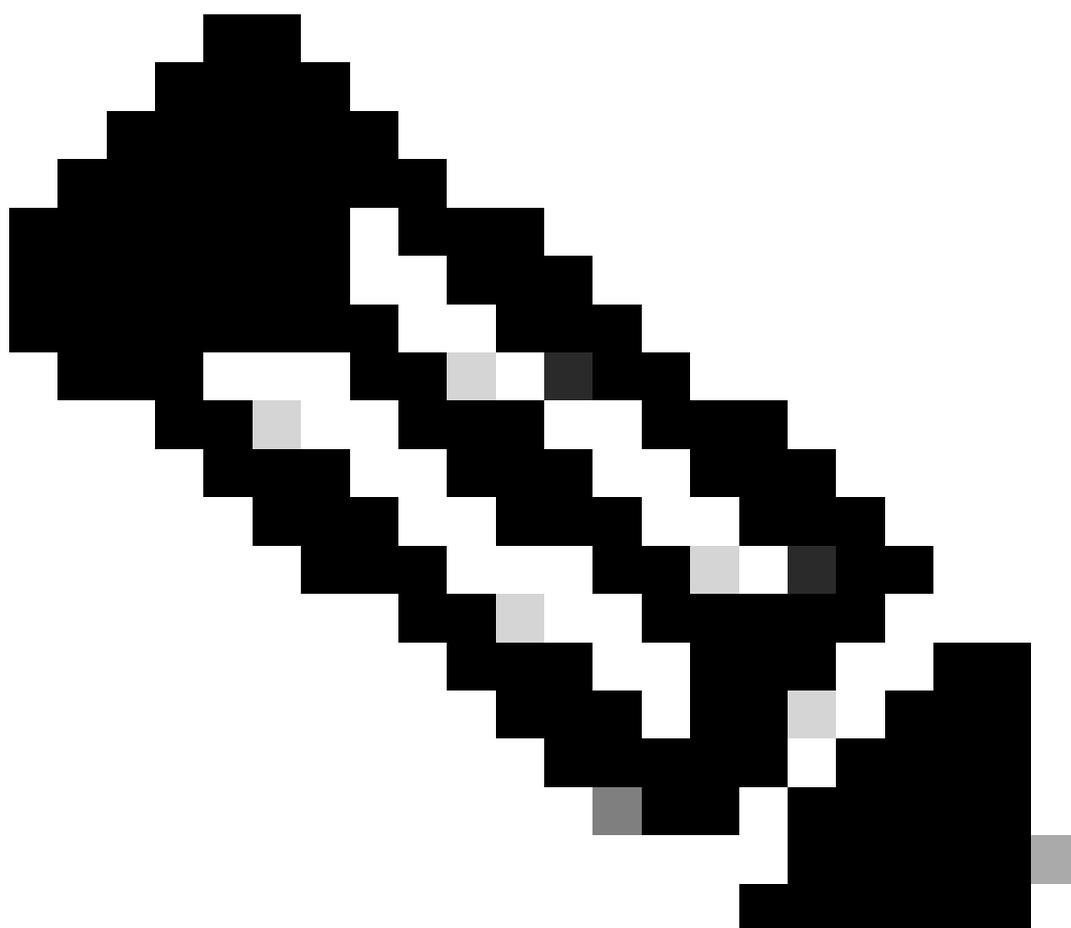
```
show process cpu platform tri
```

```
show platform software infrastructure punt
```

```
show platform hardware qfp active statistics drop
```

```
show platform hardware qfp active infra punt statistics type par cause
```

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



Remarque : Dans le fichier `punt_debug.log`, vous vous concentrez sur les indicateurs d'erreur et la grande quantité de paquets pouvant provoquer le problème.

Interface LSMPI (Linux Shared Memory Punt Interface)

Ce composant est utilisé pour transmettre des paquets et des messages du processeur de transfert au processeur de routage.

Le contrôleur Punt

Le régulateur de point est un mécanisme de protection du plan de contrôle qui permet au système de protéger et de contrôler les paquets du plan de contrôle.

Avec la commande `show platform software punt-policer`, vous pouvez voir les paquets conformes et les paquets abandonnés en raison de ce policer.

```
----- 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 --

La commande `show platform software infrastructure punt` montre les données du compteur sur les causes 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:

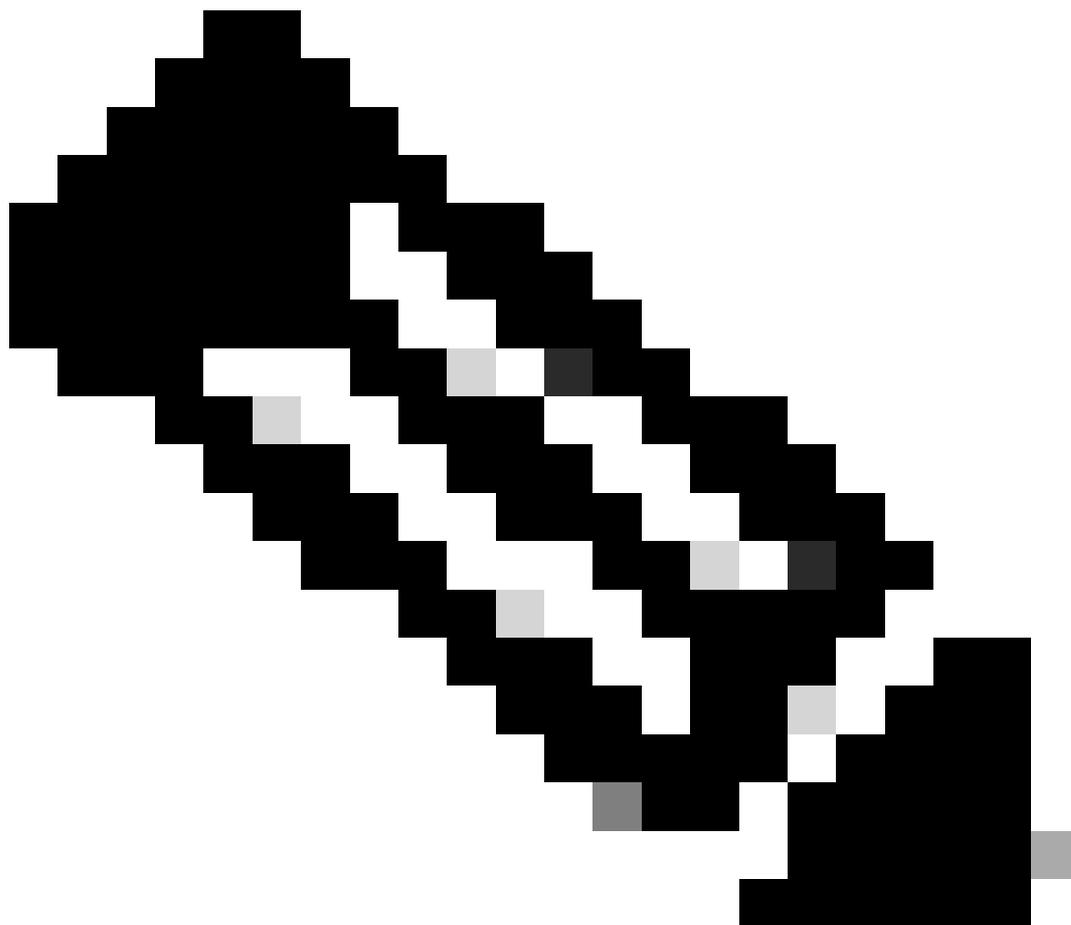
Pak-Size	In-Count	Out-Count
0+:	51108211	51144723
500+:	22069	2632
1000+:	2172	0
1500+:	3170	0

Ces données sont pertinentes pour comprendre ce qui peut avoir un impact sur le chemin punt keep alive.

Gestionnaire d'événements intégré (EEM) pour la collecte de données

Si le fichier punt_debug.log ne fournit pas suffisamment de données pour diagnostiquer le problème, le script EEM peut être utilisé pour obtenir plus de points de données au moment du problème.

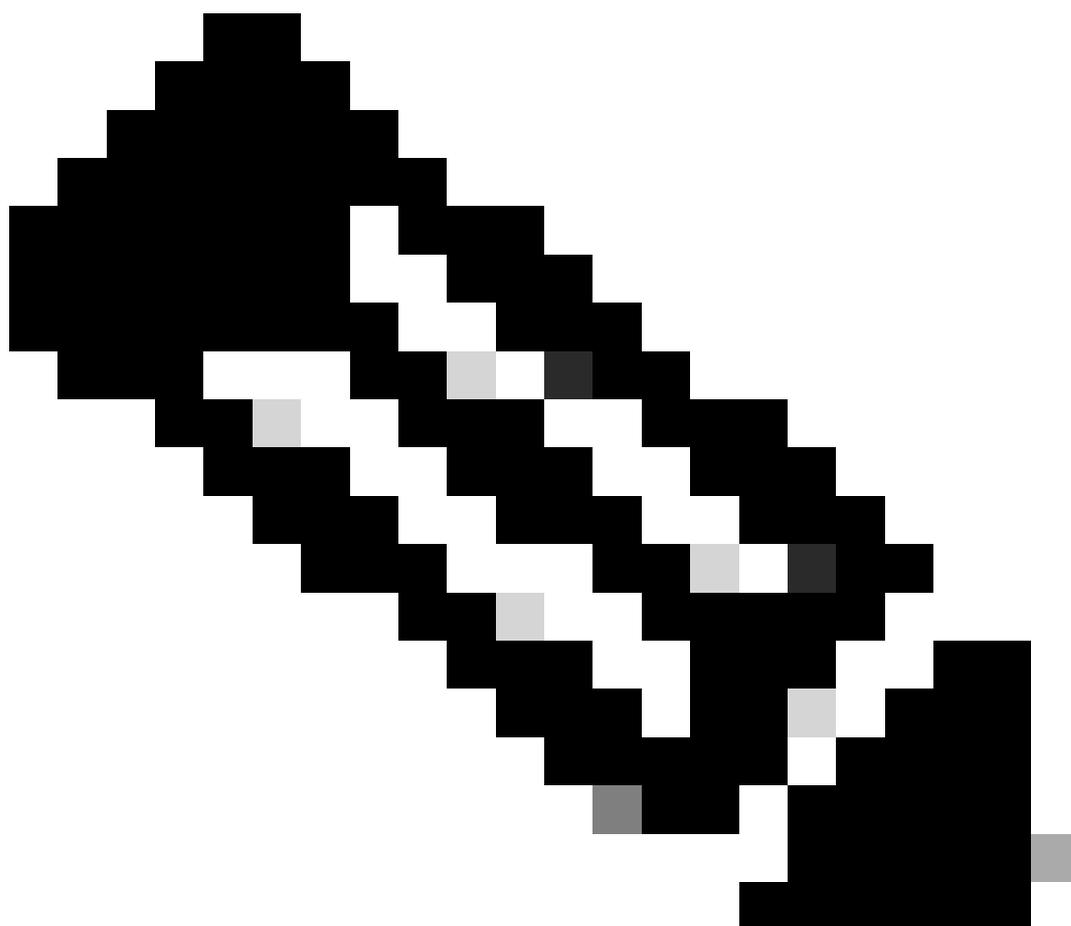
```
event manager applet punt_script authorization bypass
event syslog pattern "IOSXE_INFRA-4-NO_PUNT_KEEPLIVE" maxrun 1000
action 0.0 cli command "enable"
action 0.1 set i "0"
action 0.2 cli command "test platform software punt-keepalive ignore-fault"
action 0.3 while $i lt 10
action 0.4 syslog msg "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"
```



Remarque : Les commandes contenues dans le script varient en fonction de la plateforme sur laquelle il est configuré.

Ce script vous permet de comprendre lsmapi, les ressources et l'état punt pendant le problème.

Le script EEM inclut la commande debug platform hardware qfp active datapath crashdump qui génère le core dump qfp, nécessaire à l'équipe de développeurs et au TAC.



Remarque : Si vous soumettez un dossier au TAC Cisco, veuillez fournir le fichier de base généré par le script.

Si une trace de paquet est nécessaire, cette modification peut être ajoutée au script :

Commencez par configurer la trace des paquets, ce qui peut être fait à partir du script EEM :

```
debug platform packet-trace packet 8192 fia-trace circulaire  
debug platform condition both  
debug platform packet-trace copy packet les deux L2
```

Ensuite, démarrez et arrêtez-le avec les actions suivantes dans le script EEM :

```
action 6.2 commande cli "debug platform condition start"  
action 6.3 attente 8  
action 6.4 commande cli "debug platform condition stop"
```

Ensuite, copiez les données avec ces commandes dans un fichier séparé :

```
action 6.5, commande cli « show platform packet-trace statistics » | append bootflash:traceAll.txt"
action 6.6 cli, commande "show platform packet-trace summary | append bootflash:traceAll.txt"
action 6.7 cli, commande "show platform packet-trace packet all decode | append
bootflash:traceAll.txt"
```

Cette logique d'actions de suivi de paquet est ajoutée juste après l'instruction de fin du cycle while dans le script EEM.

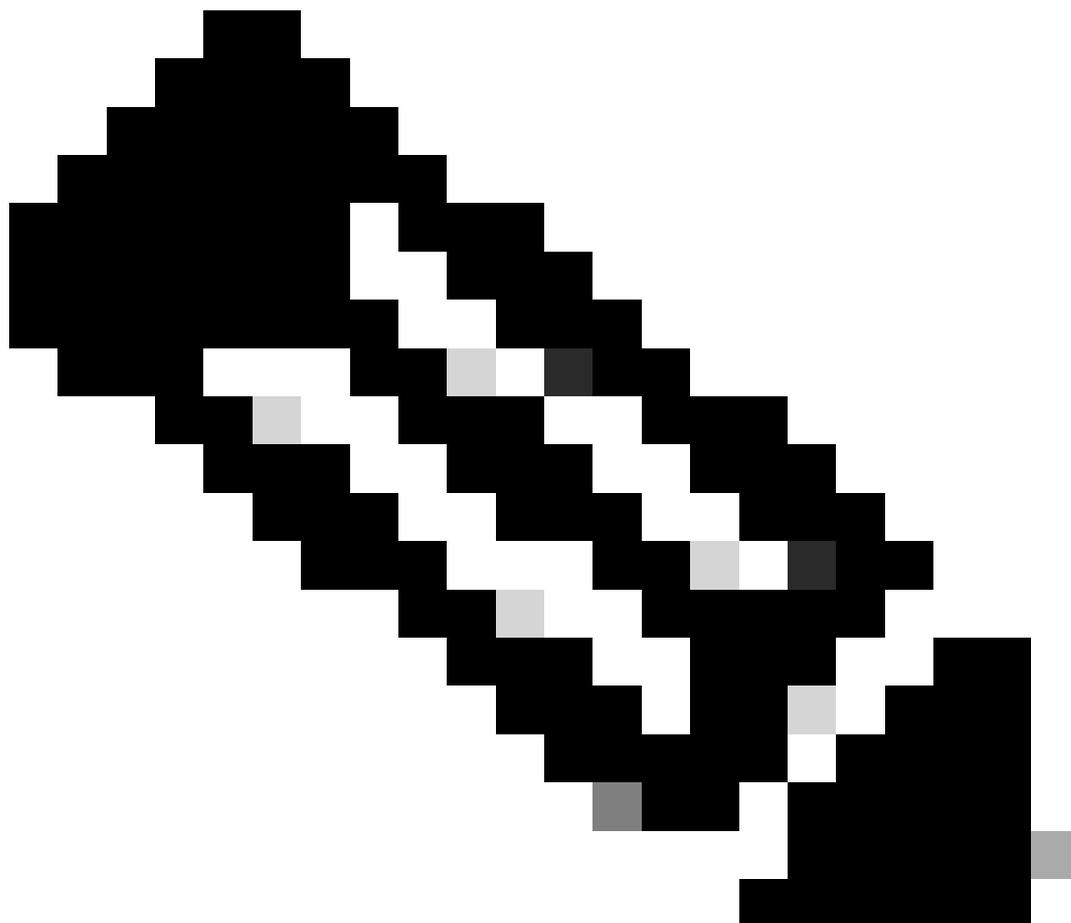
Ce script vous permet de comprendre quel type de paquets peut être à l'origine du problème.

Packet Trace est une fonctionnalité documentée dans [Dépannage avec la fonctionnalité Packet Trace de Datapath IOS XE](#)

Un exemple pratique

Un routeur CSR8000v redémarre constamment.

Après avoir extrait le rapport système, vous pouvez observer un crashdump, et un fichier de noyau iosd indiquant punt keep alive des fonctions liées dans la trace de la pile.



Remarque : Pour le décodage de la trace de la pile, une assistance TAC est requise.

Cependant, le fichier crashinfo est en texte clair et vous pouvez voir ces symptômes :

```
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

Amélioration

L'amélioration pour la génération automatique de fichier de noyau qfp a été introduite en démarrant la version 17.15 de Cisco IOS XE via l'ID de bogue Cisco [CSCwf85505](#)

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.