Dépannage avec la fonctionnalité de suivi des paquets de chemin de données IOS-XE

Table des matières

Introduction

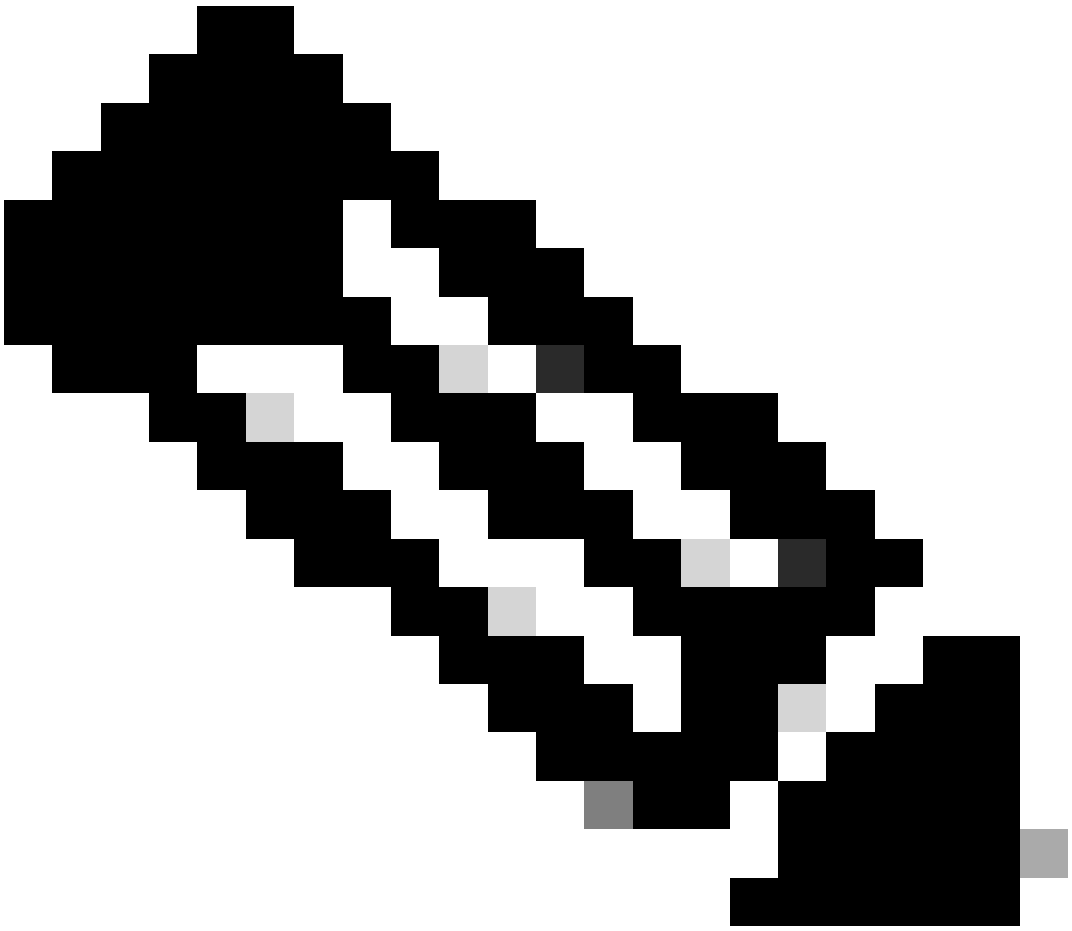
Ce document décrit comment effectuer le suivi des paquets de chemin de données pour le logiciel Cisco IOS-XE® via la fonctionnalité Packet Trace.

Conditions préalables

Exigences

Cisco vous recommande d'avoir connaissance de ces informations :

La fonctionnalité de suivi de paquets est disponible dans Cisco IOS-XE version 3.10 et versions ultérieures sur les plates-formes de routage basées sur QFP (Quantum Flow Processor), qui incluent les routeurs ASR1000, ISR4000, ISR1000, Catalyst 1000, Catalyst 8000, CSR1000v et Catalyst 8000v. Cette fonctionnalité n'est pas prise en charge sur les routeurs de services d'agrégation de la gamme ASR900 ou les commutateurs de la gamme Catalyst qui exécutent le logiciel Cisco IOS-XE.



> Remarque : la fonctionnalité de suivi de paquets ne fonctionne pas sur l'interface de gestion dédiée, GigabitEthernet0, sur les routeurs de la gamme ASR1000, car les paquets transférés sur cette interface ne sont pas traités par le QFP.

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Logiciel Cisco IOS-XE version 3.10S (15.3(3)S) et ultérieure
- Routeur ASR1000

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

Afin d'identifier des problèmes tels qu'une mauvaise configuration, une surcharge de capacité ou même un bogue logiciel ordinaire lors du dépannage, il est nécessaire de comprendre ce qui arrive à un paquet dans un système. La fonctionnalité Packet Trace de Cisco IOS-XE répond à ce besoin. Il fournit une méthode de gestion sécurisée des champs qui est utilisée pour la comptabilité et pour capturer les détails du processus par paquet en fonction d'une classe de conditions définies par l'utilisateur.

Topologie de référence

Ce schéma illustre la topologie utilisée pour les exemples décrits dans ce document :

Suivi des paquets en cours d'utilisation

Afin d'illustrer l'utilisation de la fonctionnalité de suivi de paquets, l'exemple qui est utilisé tout au long de cette section décrit un suivi du trafic ICMP (Internet Control Message Protocol) de la station de travail locale 172.16.10.2 (derrière l'ASR1K) vers l'hôte distant 172.16.20.2 dans le sens d'entrée sur l'interface GigabitEthernet0/0/1 sur l'ASR1K.

Vous pouvez suivre les paquets sur l'ASR1K en procédant comme suit :

1. Activez les débogages conditionnels de la plate-forme afin de sélectionner les paquets ou le trafic que vous voulez suivre sur l'ASR1K.

2. Activez le suivi des paquets de la plate-forme avec l'option de suivi path-trace ou Feature Invocation Array (FIA).

Guide de démarrage rapide

Voici un guide de démarrage rapide si vous connaissez déjà le contenu de ce document et si vous souhaitez consulter une section rapide sur l'interface de ligne de commande. Ce ne sont que

quelques exemples pour illustrer l'utilisation de l'outil. Reportez-vous aux sections suivantes qui traitent des syntaxes en détail et assurez-vous d'utiliser la configuration appropriée à vos besoins.

1. Configuration des conditions de plate-forme :

<#root>

debug platform condition ipv4 10.0.0.1/32 both

--> matches in and out packets with source
 or destination as 10.0.0.1/32

debug platform condition ipv4 access-list 198 egress

--> (Ensure access-list 198 is
 defined prior to configuring this command) - matches egress packets corresponding
 to access-list 198

debug platform condition interface gig 0/0/0 ingress

--> matches all ingress packets
 on interface gig 0/0/0

debug platform condition mpls 10 1 ingress

--> matches MPLS packets with top ingress
 label 10

debug platform condition ingress

--> matches all ingress packets on all interfaces
 (use cautiously)

Une fois qu'une condition de plate-forme est configurée, démarrez-la avec cette commande CLI :

<#root>

debug platform condition start

2. Configurer la trace des paquets :

<#root>

debug platform packet-trace packet 1024

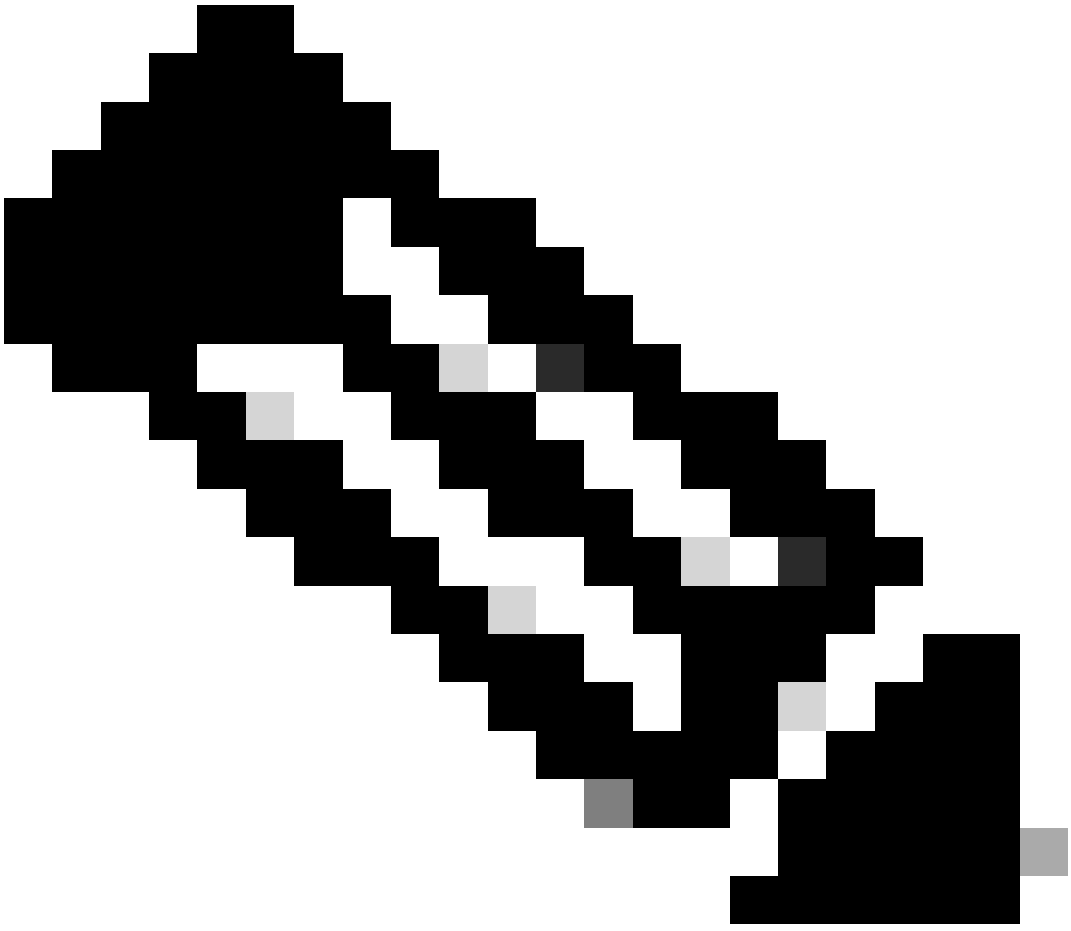
```
 -> basic path-trace, and automatically stops
tracing packets after 1024 packets. You can use "circular" option if needed
```

debug platform packet-trace packet 1024 fia-trace -

```
> enables detailed fia trace, stops
 tracing packets after 1024 packets
```

debug platform packet-trace drop [code <dropcode>]

```
 -> if you want to trace/capture only
packets that are dropped. Refer to Drop Trace section for more details.
```



Remarque : dans les versions antérieures de Cisco IOS-XE 3.x, la commande debug platform packet-trace enable est également requise pour démarrer la fonctionnalité packet-trace. Cette opération n'est plus nécessaire dans les versions 16.x de Cisco IOS-XE.

Entrez cette commande afin d'effacer la mémoire tampon de trace et de réinitialiser packet-trace :

<#root>

clear platform packet-trace statistics

  --> clear the packet trace buffer

La commande permettant d'effacer les conditions de la plate-forme et la configuration de trace de paquets est :

<#root>

clear platform condition all

  --> clears both platform conditions and the packet trace configuration

Commandes show

Vérifiez la condition de la plate-forme et la configuration du suivi des paquets après avoir appliqué les commandes précédentes afin de vous assurer que vous disposez de ce dont vous avez besoin.

<#root>

show platform conditions

 --> shows the platform conditions configured

show platform packet-trace configuration

 --> shows the packet-trace configurations

show debugging

 --> this can show both platform conditions and platform packet-trace configured

Voici les commandes permettant de vérifier les paquets tracés/capturés :

<#root>

show platform packet-trace statistics

--> statistics of packets traced

```
show platform packet-trace summary

  --> summary of all the packets traced, with input and
 output interfaces, processing result and reason.


show platform packet-trace packet 12

 -> Display path trace of FIA trace details for the 12th packet in the trace buffer
```

Activer les débogages conditionnels de plateforme

La fonctionnalité Packet Trace s’appuie sur l’infrastructure de débogage conditionnel afin de déterminer les paquets à tracer. L'infrastructure de débogage conditionnel offre la possibilité de filtrer le trafic en fonction des éléments suivants :

- Protocol
- Adresse IP et masque
- Liste de contrôle d'accès (ACL)
- Interface
- Direction du trafic (entrée ou sortie)

Ces conditions définissent où et quand les filtres sont appliqués à un paquet.

Pour le trafic utilisé dans cet exemple, activez les débogages conditionnels de plate-forme dans la direction d'entrée pour les paquets ICMP de 172.16.10.2 à 172.16.20.2. En d'autres termes, sélectionnez le trafic que vous souhaitez tracer. Il existe différentes options que vous pouvez utiliser afin de sélectionner ce trafic.

<#root>

```
ASR1000#

debug platform condition

 ?
  egress     Egress only debug
  feature    For a specific feature
  ingress    Ingress only debug
  interface  Set interface for conditional debug
  ipv4       Debug IPv4 conditions
  ipv6       Debug IPv6 conditions
  start      Start conditional debug
  stop       Stop conditional debug
```

Dans cet exemple, une liste d'accès est utilisée afin de définir la condition, comme indiqué ici :

<#root>

```
ASR1000#
```

```
show access-list 150


Extended IP access list 150
 10 permit icmp host 172.16.10.2 host 172.16.20.2
ASR1000#
```

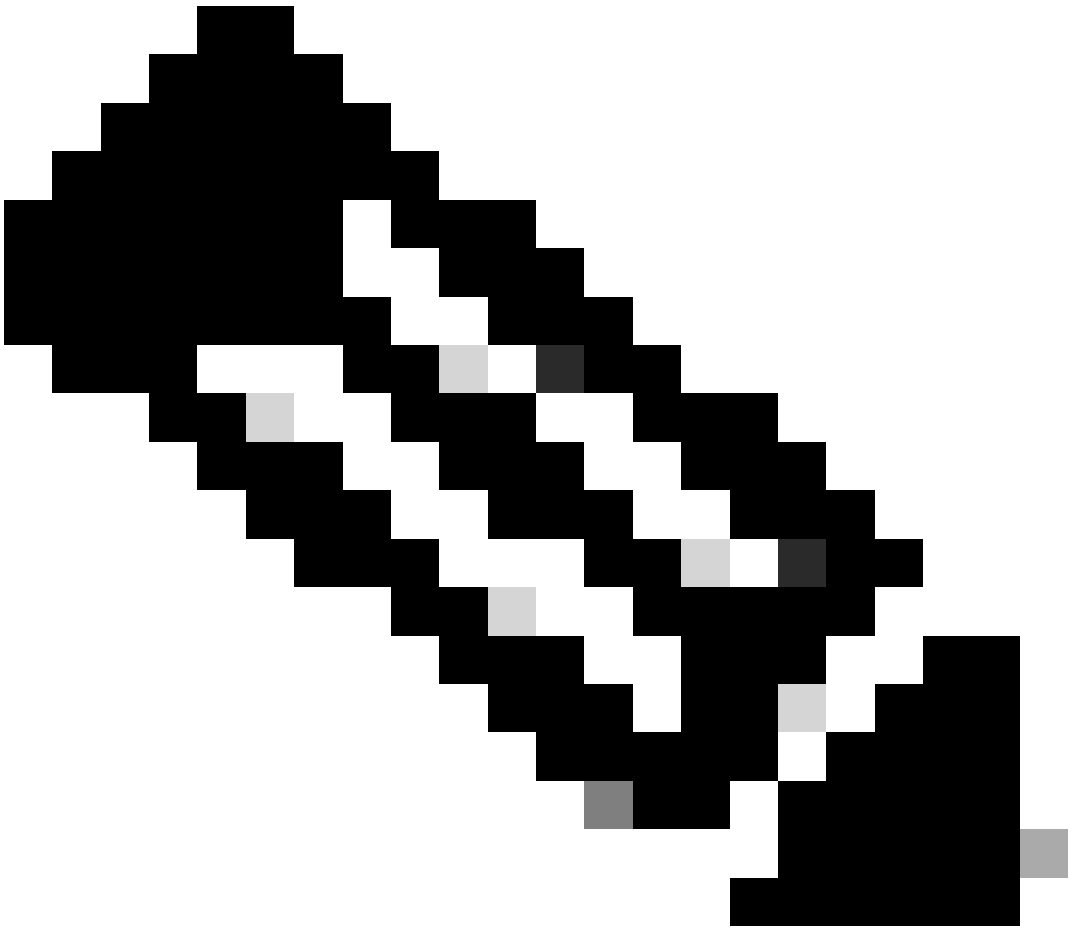
```
debug platform condition interface gig 0/0/1 ipv4
 access-list 150 ingress
```

Afin de commencer le débogage conditionnel, entrez cette commande :

<#root>

```
ASR1000#
```

```
debug platform condition start
```



Remarque : pour arrêter ou désactiver l'infrastructure de débogage conditionnel, entrez la commande debug platform condition stop.

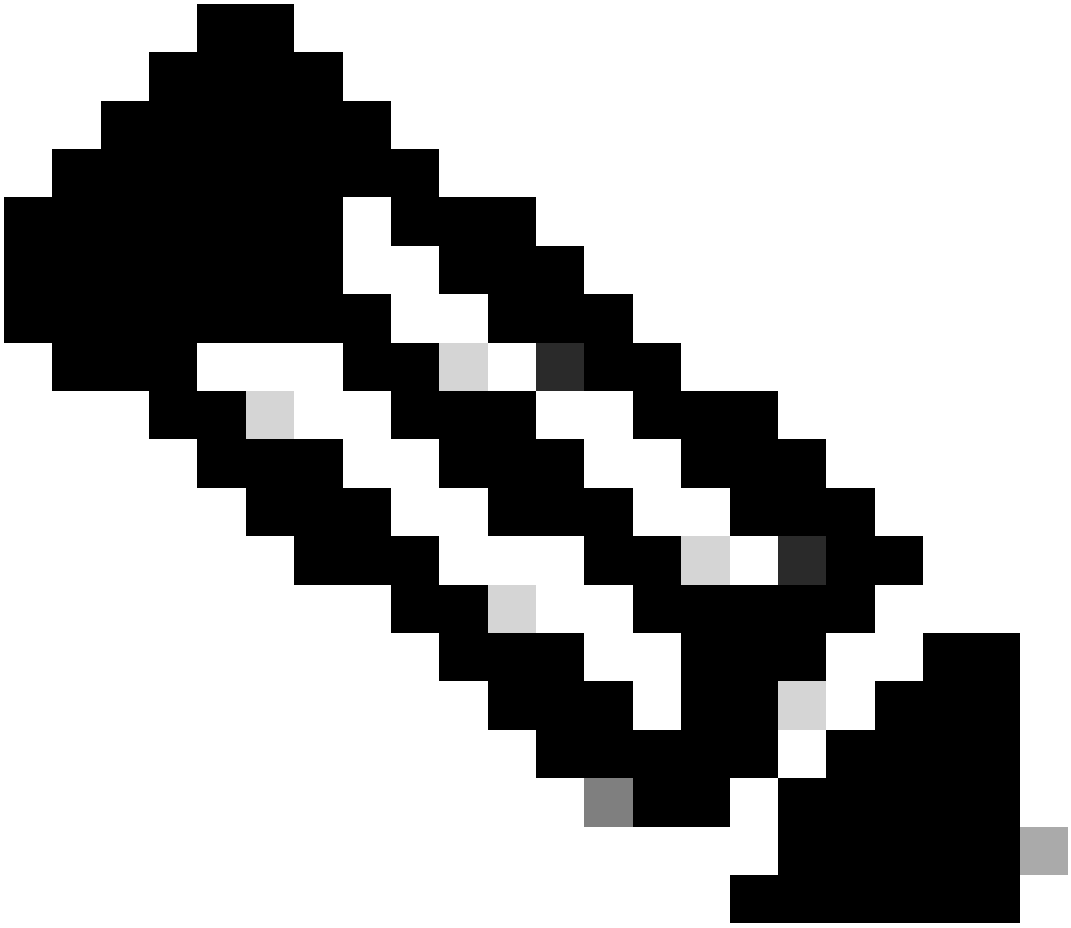
Afin d'afficher les filtres de débogage conditionnels qui sont configurés, entrez cette commande :

```
<#root>

ASR1000#

show platform conditions


Conditional Debug Global State:

Start


Conditions                                                              Direction
------------------------------------------------------------------------|---------
GigabitEthernet0/0/1                      & IPV4 ACL [150]               ingress


Feature Condition        Format                  Value
-----------------------|-----------------------|--------------------------------

ASR1000#
```

En résumé, cette configuration a été appliquée jusqu'à présent :

```
<#root>

access-list 150 permit icmp host 172.16.10.2 host 172.16.20.2

debug platform condition interface gig 0/0/1 ipv4 access-list 150 ingress
debug platform condition start
```

Activer Packet Trace



Remarque : cette section décrit en détail les options de paquet et de copie, et les autres options sont décrites plus loin dans le document.

Les traces de paquets sont prises en charge sur les interfaces physiques et logiques, telles que les interfaces de tunnel ou d'accès virtuel.

Voici la syntaxe CLI de suivi des paquets :

<#root>

```
ASR1000#
debug platform packet-trace
 ?
  copy    Copy packet data
  drop    Trace drops only
  inject  Trace injects only
  packet  Packet count
  punt    Trace punts only
```

<#root>

```
debug platform packet-trace packet <pkt-size/pkt-num> [fia-trace | summary-only]
 [circular] [data-size <data-size>]
```

Voici les descriptions des mots-clés de cette commande :

- pkt-num - Le numéro de paquet spécifie le nombre maximal de paquets qui sont conservés en même temps.
- summary-only : indique que seules les données récapitulatives sont capturées. La valeur par défaut est de capturer à la fois les données récapitulatives et les données de chemin de fonction.
- fia-trace : exécute éventuellement une trace FIA en plus des informations de données de chemin.
- data-size : permet de spécifier la taille du tampon de données de chemin, de 2 048 à 16 384 octets. La valeur par défaut est 2 048 octets.

<#root>

```
debug platform packet-trace copy packet {in | out | both} [L2 | L3 | L4]
 [size <num-bytes>]
```

Voici les descriptions des mots-clés de cette commande :

- in/out : indique la direction du flux de paquets à copier (entrée et/ou sortie).
- L2/L3/L4 : permet de spécifier l'emplacement de début de la copie du paquet. L'emplacement par défaut est la couche 2.
- size : permet de spécifier le nombre maximal d'octets copiés. La valeur par défaut est 64 octets.

Pour cet exemple, il s'agit de la commande utilisée afin d'activer la trace de paquet pour le trafic qui est sélectionné avec l'infrastructure de débogage conditionnelle :

<#root>

```
ASR1000#
debug platform packet-trace packet 16
```

Afin d'examiner la configuration de trace de paquet, entrez cette commande :

<#root>

ASR1000#

show platform packet-trace configuration

debug platform packet-trace packet 16 data-size 2048

Vous pouvez également entrer la commande show debugging afin d'afficher à la fois les débogages conditionnels de plate-forme et les configurations de trace de paquet :

<#root>

ASR1000#

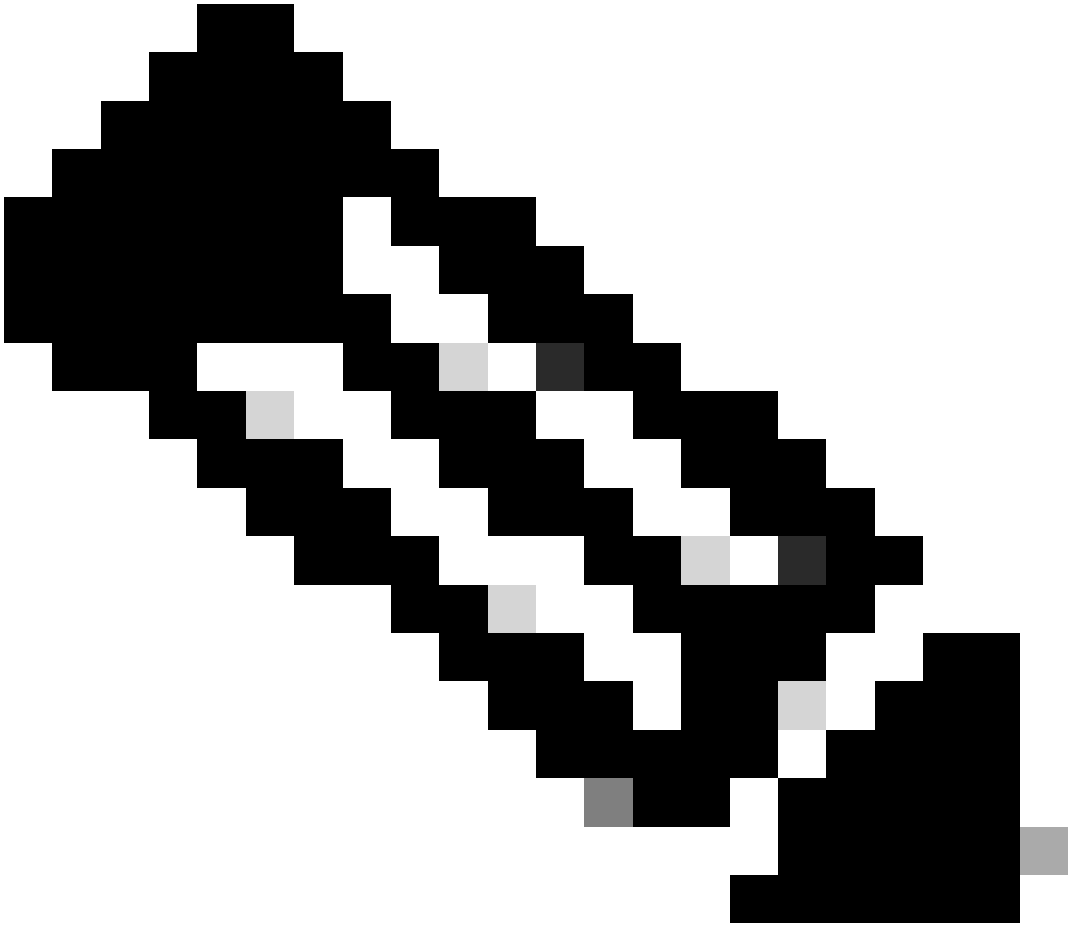
show debugging

IOSXE Conditional Debug Configs:

Conditional Debug Global State: Start

Conditions

```
                                                                Direction
------------------------------------------------------------------------|---------
GigabitEthernet0/0/1                     & IPV4 ACL [150]              ingress
...
IOSXE Packet Tracing Configs:

Feature Condition        Format                  Value
----------------------|----------------------|--------------------------------

Feature  Type           Submode                                          Level
-------|--------------|-----------------------------------------------|----------

IOSXE Packet Tracing Configs:

debug platform packet-trace packet 16 data-size 2048
```



Remarque : entrez la commande clear platform condition all afin d'effacer toutes les conditions de débogage de la plate-forme et les configurations et données de trace de paquet.

En résumé, ces données de configuration ont été utilisées jusqu'à présent afin d'activer packet-trace :

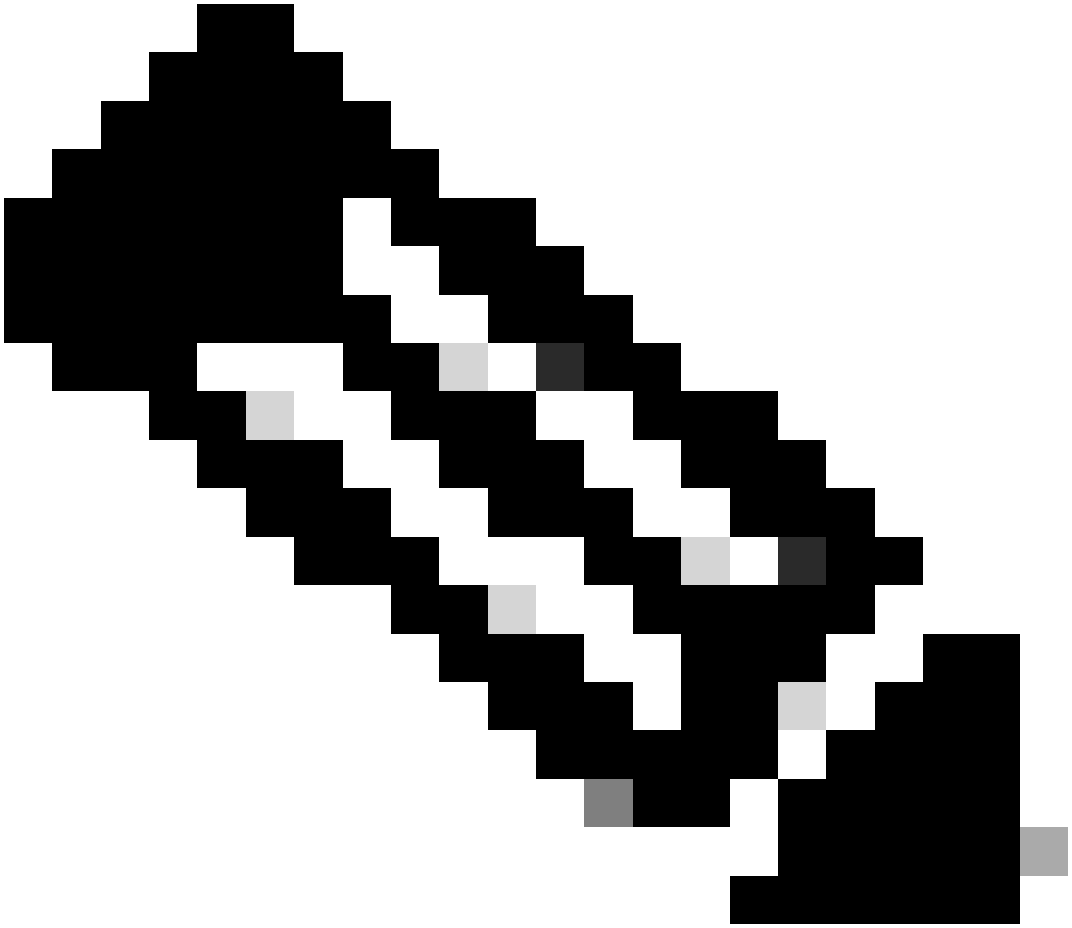
<#root>

debug platform packet-trace packet 16

Limitation des conditions de sortie avec Packet Traces

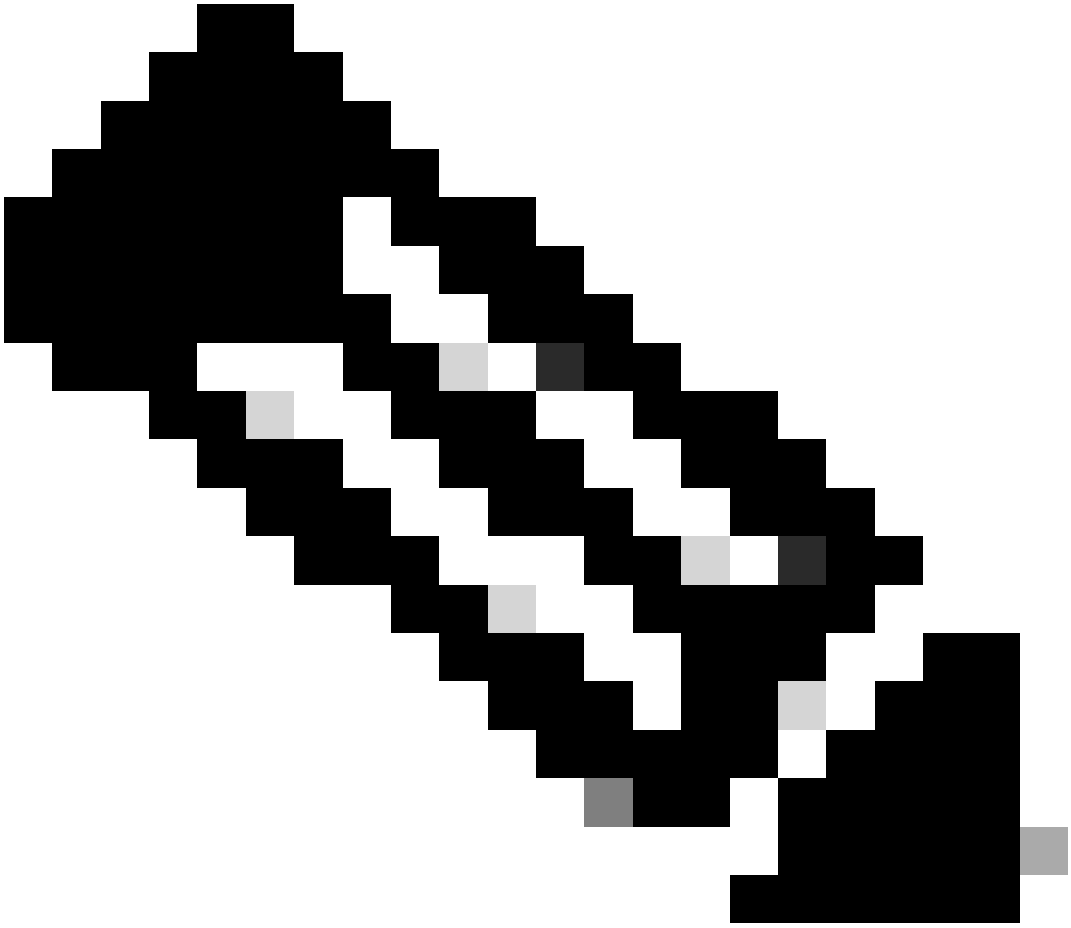
Les conditions définissent les filtres conditionnels et quand ils sont appliqués à un paquet. Par exemple, debug platform condition interface g0/0/0 egress signifie qu'un paquet est identifié

comme une correspondance lorsqu'il atteint la sortie FIA sur l'interface g0/0/0, de sorte que tout traitement de paquet qui a lieu depuis l'entrée jusqu'à ce point est manqué.



> Remarque : Cisco vous recommande vivement d'utiliser des conditions d'entrée pour les suivis de paquets afin d'obtenir les données les plus complètes et les plus significatives possibles. Les conditions de sortie peuvent être utilisées, mais soyez conscient des limitations.

Afficher les résultats de Packet Trace



Remarque : cette section suppose que path-trace est activé.

Trois niveaux spécifiques d’inspection sont fournis par le suivi de paquets :

- Gestion de comptes
- Résumé par paquet
- Données de chemin par paquet

Lorsque cinq paquets de requête ICMP sont envoyés de 172.16.10.2 à 172.16.20.2, ces commandes peuvent être utilisées afin d'afficher les résultats de suivi de paquets :

<#root>

ASR1000#

show platform packet-trace statistics

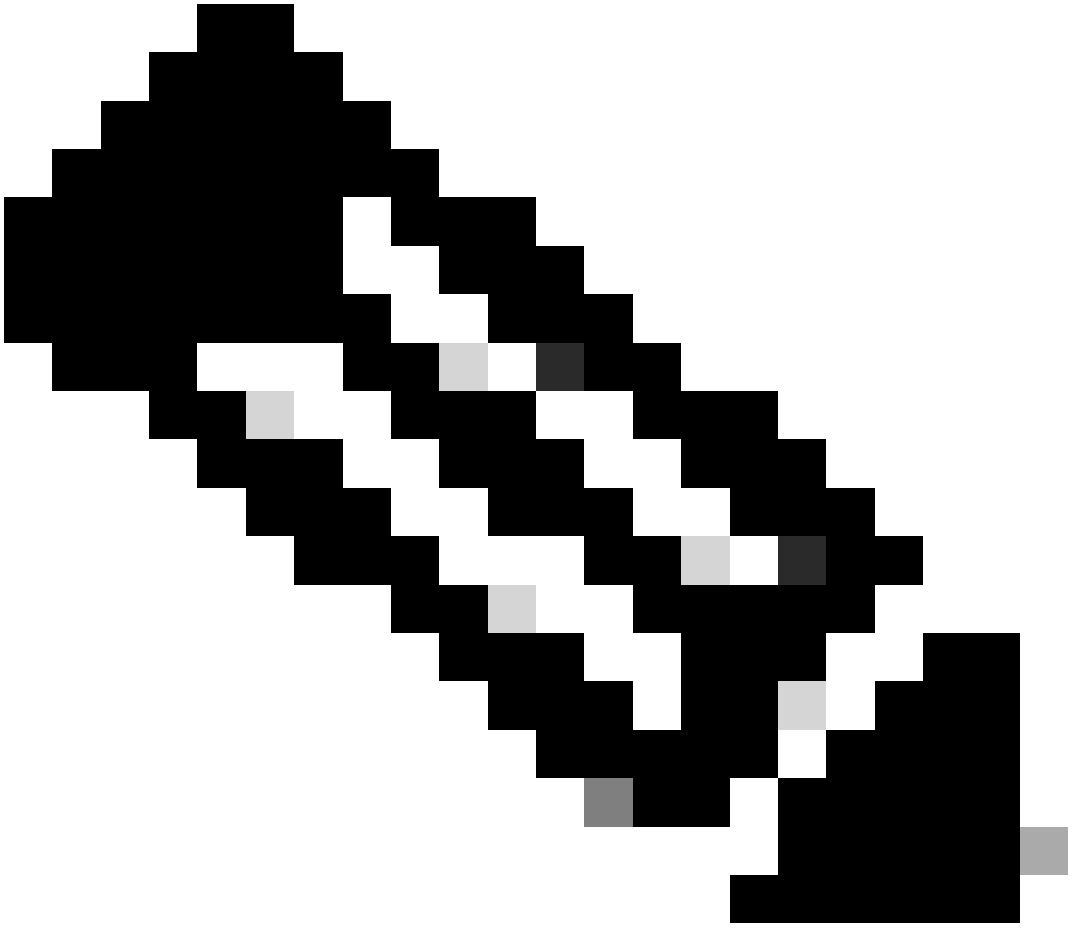
```
Packets Traced: 5


 Ingress  5
 Inject   0
 Forward  5
 Punt     0
 Drop     0
 Consume  0

 ASR1000#
```

```
show platform packet-trace summary
```

```
Pkt

   Input            Output           State  Reason


0

     Gi0/0/1          Gi0/0/0          FWD
 1     Gi0/0/1          Gi0/0/0          FWD
 2     Gi0/0/1          Gi0/0/0          FWD
 3     Gi0/0/1          Gi0/0/0          FWD
 4     Gi0/0/1          Gi0/0/0          FWD

 ASR1000#
```

```
show platform packet-trace packet 0
```

```
 Packet: 0

          CBUG ID: 4
 Summary
 Input     : GigabitEthernet0/0/1
 Output    : GigabitEthernet0/0/0
 State     : FWD
 Timestamp
    Start   : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
    Stop    : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
 Path Trace
 Feature: IPV4
 Source      : 172.16.10.2
 Destination : 172.16.20.2
 Protocol    : 1 (ICMP)

 ASR1000#
```



Remarque : la troisième commande fournit un exemple qui illustre comment afficher le suivi des paquets pour chaque paquet. Dans cet exemple, le premier paquet tracé est représenté.

À partir de ces sorties, vous pouvez voir que cinq paquets sont tracés et que vous pouvez voir l'interface d'entrée, l'interface de sortie, l'état et la trace de chemin.

Province	Faire Remarquer
FWD	Le paquet est planifié/mis en file d'attente pour être transmis au saut suivant via une interface de sortie.
FAIRE LA MOUCHE	Le paquet est envoyé du processeur de transfert (FP) au processeur de routage (RP) (plan de contrôle).
CHUTE	Le paquet est abandonné sur le FP. Exécutez FIA trace, utilisez des compteurs de dépôt globaux ou utilisez des débogages de chemin de données afin de trouver plus de détails pour les raisons de dépôt.
POINTS	Le paquet est consommé au cours d'un processus de paquet, par exemple

NÉGATIFS	pendant la requête ping ICMP ou les paquets de chiffrement.

Les compteurs d'entrée et d'injection dans la sortie de statistiques de suivi de paquets correspondent respectivement aux paquets qui entrent par une interface externe et aux paquets qui sont vus comme injectés depuis le plan de contrôle.

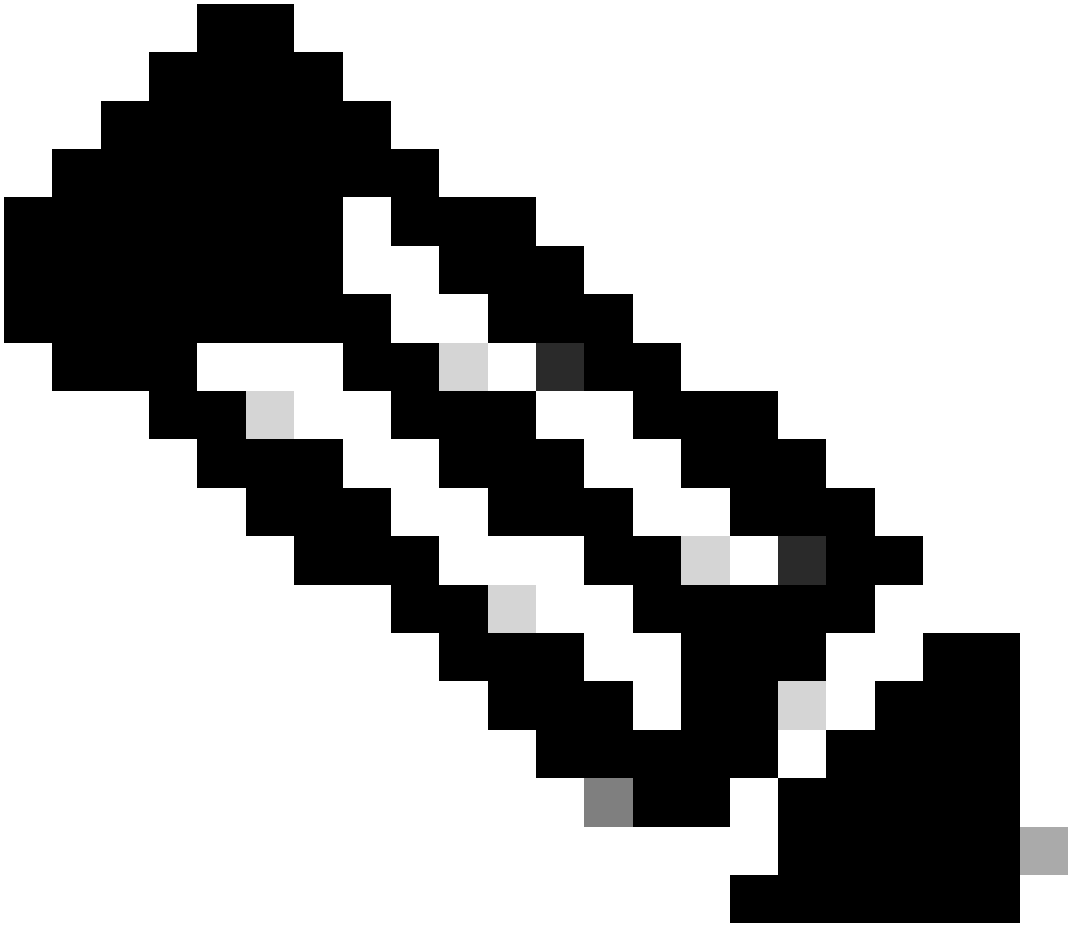
FIA Trace

Le FIA contient la liste des fonctionnalités exécutées séquentiellement par les moteurs de processeur de paquets (PPE) dans le processeur de flux quantique (QFP) lorsqu'un paquet est transféré en entrée ou en sortie. Les fonctions sont basées sur les données de configuration appliquées à la machine. Ainsi, une trace FIA aide à comprendre le flux du paquet à travers le système pendant le traitement du paquet.

Vous devez appliquer ces données de configuration afin d'activer le suivi des paquets avec FIA :

```
<#root>

ASR1000#

debug platform packet-trace packet 16 fia-trace
```

Afficher les résultats de Packet Trace



> Remarque : cette section suppose que le suivi FIA est activé. En outre, lorsque vous ajoutez ou modifiez les commandes de suivi de paquets actuelles, les détails de suivi de paquets mis en mémoire tampon sont effacés, de sorte que vous devez renvoyer du trafic afin de pouvoir le suivre.

Envoyez cinq paquets ICMP de 172.16.10.2 à 172.16.20.2 après avoir entré la commande utilisée afin d'activer la trace FIA, comme décrit dans la section précédente.

<#root>

ASR1000#

show platform packet-trace summary

```
Pkt   Input           Output          State  Reason
0     Gi0/0/1         Gi0/0/0         FWD
1     Gi0/0/1         Gi0/0/0         FWD
2     Gi0/0/1         Gi0/0/0         FWD
3     Gi0/0/1         Gi0/0/0         FWD
```

```
4     Gi0/0/1          Gi0/0/0          FWD

ASR1000#
```

show platform packet-trace packet 0

```
Packet: 0           CBUG ID: 9
Summary
  Input     : GigabitEthernet0/0/1
  Output    : GigabitEthernet0/0/0
  State     : FWD
  Timestamp
    Start   : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
    Stop    : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
Path Trace
  Feature: IPV4
    Source      : 172.16.10.2
    Destination : 172.16.20.2
    Protocol    : 1 (ICMP)
  Feature: FIA_TRACE
    Entry     : 0x8059dbe8 - DEBUG_COND_INPUT_PKT
    Timestamp : 3685243309297
  Feature: FIA_TRACE
    Entry     : 0x82011a00 - IPV4_INPUT_DST_LOOKUP_CONSUME
    Timestamp : 3685243311450
  Feature: FIA_TRACE
    Entry     : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
    Timestamp : 3685243312427
  Feature: FIA_TRACE
    Entry     : 0x82004b68 - IPV4_OUTPUT_LOOKUP_PROCESS
    Timestamp : 3685243313230
  Feature: FIA_TRACE
    Entry     : 0x8034f210 - IPV4_INPUT_IPOPTIONS_PROCESS
    Timestamp : 3685243315033
  Feature: FIA_TRACE
    Entry     : 0x82013200 - IPV4_OUTPUT_GOTO_OUTPUT_FEATURE
    Timestamp : 3685243315787
  Feature: FIA_TRACE
    Entry     : 0x80321450 - IPV4_VFR_REFRAG
    Timestamp : 3685243316980
  Feature: FIA_TRACE
    Entry     : 0x82014700 - IPV6_INPUT_L2_REWRITE
    Timestamp : 3685243317713
  Feature: FIA_TRACE
    Entry     : 0x82000080 - IPV4_OUTPUT_FRAG
    Timestamp : 3685243319223
  Feature: FIA_TRACE
    Entry     : 0x8200e500 - IPV4_OUTPUT_DROP_POLICY
    Timestamp : 3685243319950
  Feature: FIA_TRACE
    Entry     : 0x8059aff4 - PACTRAC_OUTPUT_STATS
    Timestamp : 3685243323603
  Feature: FIA_TRACE
    Entry     : 0x82016100 - MARMOT_SPA_D_TRANSMIT_PKT
    Timestamp : 3685243326183

ASR1000#
```

Vérifier la FIA associée à une interface

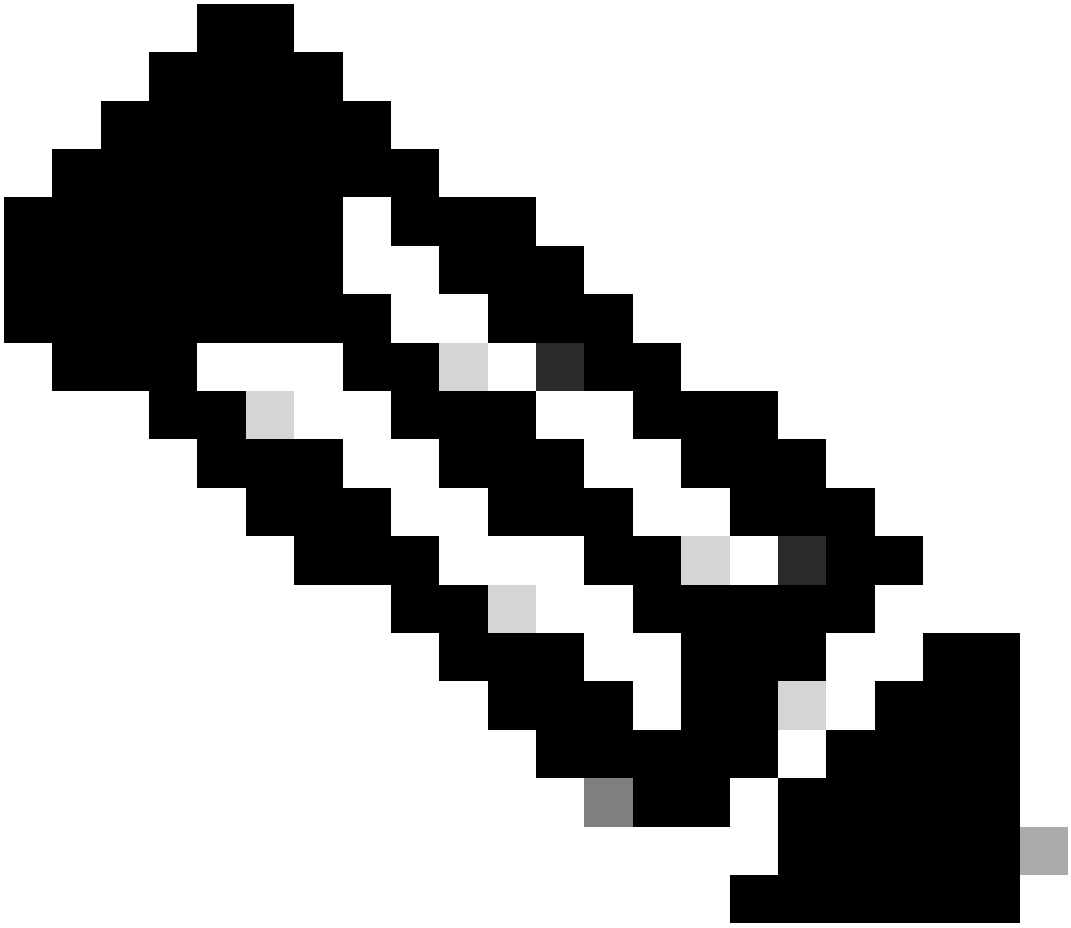
Lorsque vous activez les débogages conditionnels de plate-forme, le débogage conditionnel est ajouté à la FIA en tant que fonctionnalité. En fonction de l'ordre de traitement des fonctions sur l'interface, le filtre conditionnel doit être défini en conséquence, par exemple, si l'adresse pré- ou post-NAT doit être utilisée dans le filtre conditionnel.

Ce résultat montre l'ordre des fonctionnalités dans la FIA pour le débogage conditionnel de plate-forme qui est activé dans la direction d'entrée :

<#root>

```
ASR1000#

show platform hardware qfp active interface if-name GigabitEthernet 0/0/1


 General interface information
 Interface Name: GigabitEthernet0/0/1
 Interface state: VALID
 Platform interface handle: 10
 QFP interface handle: 8
 Rx uidb: 1021
 Tx uidb: 131064
 Channel: 16
 Interface Relationships

 BGPPA/QPPB interface configuration information
 Ingress: BGPPA/QPPB not configured. flags: 0000
 Egress : BGPPA not configured. flags: 0000

 ipv4_input enabled.
 ipv4_output enabled.
 layer2_input enabled.
 layer2_output enabled.
 ess_ac_input enabled.

 Features Bound to Interface:
 2 GIC FIA state
 48 PUNT INJECT DB
 39 SPA/Marmot server
 40 ethernet
 1 IFM
 31 icmp_svr
 33 ipfrag_svr
 34 ipreass_svr
 36 ipvfr_svr
 37 ipv6vfr_svr
 12 CPP IPSEC
 Protocol 0 - ipv4_input
 FIA handle - CP:0x108d99cc  DP:0x8070f400
 IPV4_INPUT_DST_LOOKUP_ISSUE (M)
 IPV4_INPUT_ARL_SANITY (M)


  CBUG_INPUT_FIA
```

```
 DEBUG_COND_INPUT_PKT


IPV4_INPUT_DST_LOOKUP_CONSUME (M)
IPV4_INPUT_FOR_US_MARTIAN (M)
IPV4_INPUT_IPSEC_CLASSIFY
IPV4_INPUT_IPSEC_COPROC_PROCESS
IPV4_INPUT_IPSEC_RERUN_JUMP
IPV4_INPUT_LOOKUP_PROCESS (M)
IPV4_INPUT_IPOPTIONS_PROCESS (M)
IPV4_INPUT_GOTO_OUTPUT_FEATURE (M)
Protocol 1 - ipv4_output
FIA handle - CP:0x108d9a34  DP:0x8070eb00
IPV4_OUTPUT_VFR
MC_OUTPUT_GEN_RECYCLE (D)
IPV4_VFR_REFRAG (M)
IPV4_OUTPUT_IPSEC_CLASSIFY
IPV4_OUTPUT_IPSEC_COPROC_PROCESS
IPV4_OUTPUT_IPSEC_RERUN_JUMP
IPV4_OUTPUT_L2_REWRITE (M)
IPV4_OUTPUT_FRAG (M)
IPV4_OUTPUT_DROP_POLICY (M)
PACTRAC_OUTPUT_STATS
MARMOT_SPA_D_TRANSMIT_PKT
DEF_IF_DROP_FIA (M)
Protocol 8 - layer2_input
FIA handle - CP:0x108d9bd4  DP:0x8070c700
LAYER2_INPUT_SIA (M)
CBUG_INPUT_FIA
DEBUG_COND_INPUT_PKT
LAYER2_INPUT_LOOKUP_PROCESS (M)
LAYER2_INPUT_GOTO_OUTPUT_FEATURE (M)
Protocol 9 - layer2_output
FIA handle - CP:0x108d9658  DP:0x80714080
LAYER2_OUTPUT_SERVICEWIRE (M)
LAYER2_OUTPUT_DROP_POLICY (M)
PACTRAC_OUTPUT_STATS
MARMOT_SPA_D_TRANSMIT_PKT
DEF_IF_DROP_FIA (M)
Protocol 14 - ess_ac_input
FIA handle - CP:0x108d9ba0  DP:0x8070cb80
PPPOE_GET_SESSION
ESS_ENTER_SWITCHING
PPPOE_HANDLE_UNCLASSIFIED_SESSION
DEF_IF_DROP_FIA (M)

QfpEth Physical Information
DPS Addr: 0x11215eb8
Submap Table Addr: 0x00000000
VLAN Ethertype: 0x8100
QOS Mode: Per Link

ASR1000#
```



Remarque : CBUG_INPUT_FIA et DEBUG_COND_INPUT_PKT correspondent aux fonctionnalités de débogage conditionnel configurées sur le routeur.

Vider les paquets suivis

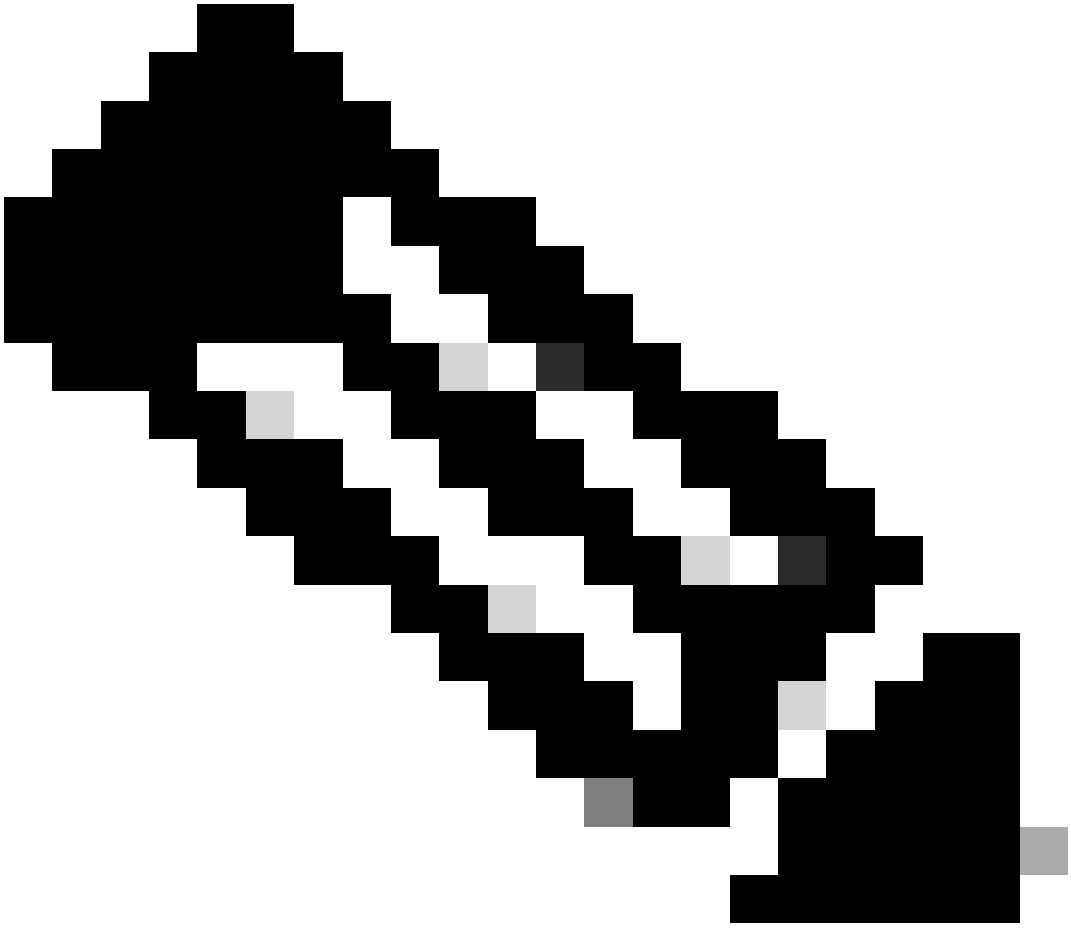
Vous pouvez copier et vider les paquets au fur et à mesure de leur traçage, comme le décrit cette section. Cet exemple montre comment copier un maximum de 2 048 octets des paquets dans la direction d'entrée (172.16.10.2 à 172.16.20.2).

Voici la commande supplémentaire qui est nécessaire :

<#root>

ASR1000#

debug platform packet-trace copy packet input size 2048



Remarque : la taille du paquet copié est comprise entre 16 et 2 048 octets.

Entrez cette commande afin de vider les paquets copiés :

<#root>

ASR1000#

show platform packet-trace packet 0

```
 Packet: 0           CBUG ID: 14
 Summary
 Input     : GigabitEthernet0/0/1
 Output    : GigabitEthernet0/0/0
 State     : FWD
 Timestamp
     Start   : 1819281992118 ns (05/17/2014 06:40:01.207240 UTC)
     Stop    : 1819282095121 ns (05/17/2014 06:40:01.207343 UTC)
 Path Trace
 Feature: IPV4
```

```
 Source      : 172.16.10.2
 Destination : 172.16.20.2
 Protocol    : 1 (ICMP)
 Feature: FIA_TRACE
 Entry     : 0x8059dbe8 - DEBUG_COND_INPUT_PKT
 Timestamp : 4458180580929

 <some content excluded>

 Feature: FIA_TRACE
 Entry     : 0x82016100 - MARMOT_SPA_D_TRANSMIT_PKT
 Timestamp : 4458180593896
```

Packet Copy In

```
 a4934c8e 33020023 33231379 08004500 00640160 0000ff01 5f16ac10 0201ac10
 01010800 1fd40024 00000000 000184d0 d980abcd abcdabcd abcdabcd abcdabcd
 abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd abcdabcd
 abcdabcd abcdabcd abcdabcd abcdabcd abcd

 ASR1000#
```

Abandonner la trace

Drop trace est disponible dans le logiciel Cisco IOS-XE versions 3.11 et ultérieures. Elle active le suivi des paquets uniquement pour les paquets abandonnés. Voici quelques points forts de cette fonctionnalité :

- Elle vous permet éventuellement de spécifier la rétention des paquets pour un code d'abandon spécifique.
- Il peut être utilisé sans conditions globales ou d'interface afin de capturer des événements d'abandon.
- Une capture d'événement de suppression signifie que seule la suppression elle-même est suivie, et non la durée de vie du paquet. Cependant, il vous permet toujours de capturer des données récapitulatives, des données de tuple et le paquet afin d'aider à affiner les conditions ou de fournir des indices à l'étape de débogage suivante.

Voici la syntaxe de commande qui est utilisée afin d'activer les traces de paquets de type drop :

<#root>

debug platform packet-trace drop [code <code-num>]

Le code d'abandon est le même que l'ID d'abandon, comme indiqué dans le résultat de la commande show platform hardware qfp active statistics drop detail :

<#root>

```
ASR1000#
show platform hardware qfp active statistics drop detail

-------------------------------------------------------------------------------
   ID
  Global Drop Stats                          Packets                  Octets
-------------------------------------------------------------------------------
   60
  IpTtlExceeded                                3                     126
    8
  Ipv4Acl                                     32                    3432
```

Exemple de scénario Drop Trace

Appliquez cette liste de contrôle d'accès sur l'interface Gig 0/0/0 de l'ASR1K afin de supprimer le trafic de 172.16.10.2 à 172.16.20.2 :

```
access-list 199 deny ip host 172.16.10.2 host 172.16.20.2
access-list 199 permit ip any any
interface Gig 0/0/0
  ip access-group 199 out
```

Une fois la liste de contrôle d'accès en place, qui abandonne le trafic de l'hôte local vers l'hôte distant, appliquez cette configuration drop-trace :

<#root>

```
debug platform condition interface Gig 0/0/1 ingress

debug platform condition start

debug platform packet-trace packet 1024 fia-trace

debug platform packet-trace drop
```

Envoyez cinq paquets de requête ICMP de 172.16.10.2 à 172.16.20.2. La commande drop trace capture les paquets abandonnés par la liste de contrôle d'accès, comme indiqué :

<#root>

ASR1000#

show platform packet-trace statistics

```
 Packets Summary
 Matched  5
 Traced   5
 Packets Received
 Ingress  5
 Inject   0
 Packets Processed
 Forward  0
 Punt     0


Drop     5
 Count       Code  Cause
 5           8     Ipv4Acl


 Consume  0


 ASR1000#
```

show platform packet-trace summary

```
 Pkt   Input           Output          State  Reason
 0     Gi0/0/1         Gi0/0/0         DROP   8   (Ipv4Acl)
 1     Gi0/0/1         Gi0/0/0         DROP   8   (Ipv4Acl)
 2     Gi0/0/1         Gi0/0/0         DROP   8   (Ipv4Acl)
 3     Gi0/0/1         Gi0/0/0         DROP   8   (Ipv4Acl)
 4     Gi0/0/1         Gi0/0/0         DROP   8   (Ipv4Acl)


 ASR1K#
```

debug platform condition stop

```


 ASR1K#
```

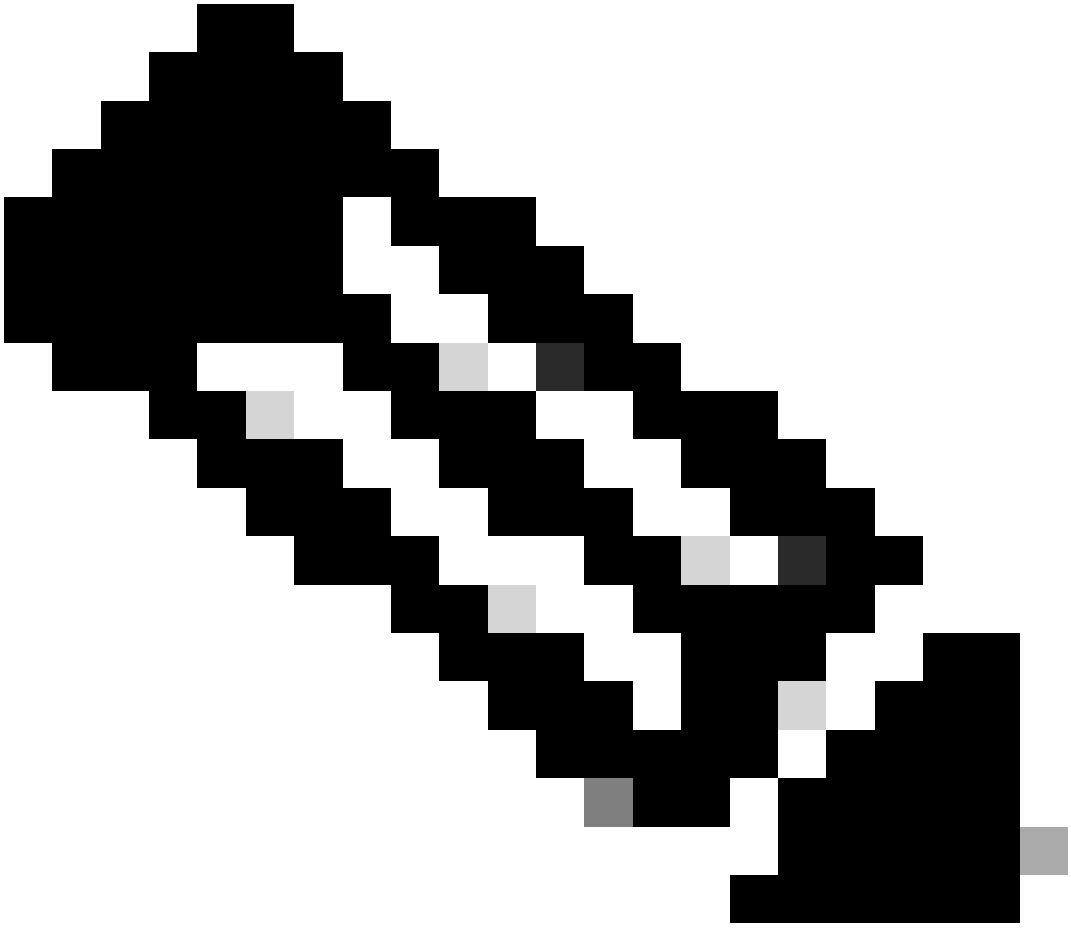
show platform packet-trace packet 0

```
 Packet: 0           CBUG ID: 140
 Summary
 Input     : GigabitEthernet0/0/1
 Output    : GigabitEthernet0/0/0


State     : DROP 8   (Ipv4Acl)


 Timestamp
 Start   : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
 Stop    : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
 Path Trace
 Feature: IPV4
 Source      : 172.16.10.2
 Destination : 172.16.20.2
```

```
Protocol    : 1 (ICMP)
Feature: FIA_TRACE
Entry      : 0x806c7eac - DEBUG_COND_INPUT_PKT
Lapsed time: 1031 ns
Feature: FIA_TRACE
Entry      : 0x82011c00 - IPV4_INPUT_DST_LOOKUP_CONSUME
Lapsed time: 657 ns
Feature: FIA_TRACE
Entry      : 0x806a2698 - IPV4_INPUT_ACL
Lapsed time: 2773 ns
Feature: FIA_TRACE
Entry      : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
Lapsed time: 1013 ns
Feature: FIA_TRACE
Entry      : 0x82004500 - IPV4_OUTPUT_LOOKUP_PROCESS
Lapsed time: 2951 ns
Feature: FIA_TRACE
Entry      : 0x8041771c - IPV4_INPUT_IPOPTIONS_PROCESS
Lapsed time: 373 ns
Feature: FIA_TRACE
Entry      : 0x82013400 - MPLS_INPUT_GOTO_OUTPUT_FEATURE
Lapsed time: 2097 ns
Feature: FIA_TRACE
Entry      : 0x803c60b8 - IPV4_MC_OUTPUT_VFR_REFRAG
Lapsed time: 373 ns
Feature: FIA_TRACE
Entry      : 0x806db148 - OUTPUT_DROP
Lapsed time: 1297 ns
Feature: FIA_TRACE
Entry      : 0x806a0c98 - IPV4_OUTPUT_ACL
Lapsed time: 78382 ns

ASR1000#
```

Injecter et poinçonner des traces

La fonctionnalité de suivi des paquets d'injection et de punt a été ajoutée dans le logiciel Cisco IOS-XE version 3.12 et ultérieure afin de suivre les paquets punt (paquets reçus sur le FP qui sont puntés sur le plan de contrôle) et d'injecter (paquets qui sont injectés sur le FP à partir du plan de contrôle).



Remarque : la commande punt trace peut fonctionner sans les conditions globales ou d'interface, tout comme une commande drop trace. Cependant, les conditions doivent être définies pour qu'une trace d'injection fonctionne.

Voici un exemple de punt et inject packet trace lorsque vous envoyez une requête ping à partir du routeur ASR1K vers un routeur adjacent :

```
<#root>

ASR1000#

debug platform condition ipv4 172.16.10.2/32 both
```

```
ASR1000#
```

debug platform condition start

```
ASR1000#
```

debug platform packet-trace punt

```
ASR1000#
```

debug platform packet-trace inject

```
ASR1000#
```

debug platform packet-trace packet 16

```
ASR1000#
ASR1000#ping 172.16.10.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 14/14/15 ms
ASR1000#
```

Vous pouvez maintenant vérifier les punt et nject trace rrésultats :

<#root>

ASR1000#

show platform packet-trace summary

Pkt	Input	Output	State	Reason
0	INJ.2	Gi0/0/1	FWD	
1	Gi0/0/1	internal0/0/rp:0	PUNT	11 (For-us data)
2	INJ.2	Gi0/0/1	FWD	
3	Gi0/0/1	internal0/0/rp:0	PUNT	11 (For-us data)
4	INJ.2	Gi0/0/1	FWD	
5	Gi0/0/1	internal0/0/rp:0	PUNT	11 (For-us data)
6	INJ.2	Gi0/0/1	FWD	
7	Gi0/0/1	internal0/0/rp:0	PUNT	11 (For-us data)
8	INJ.2	Gi0/0/1	FWD	
9	Gi0/0/1	internal0/0/rp:0	PUNT	11 (For-us data)

ASR1000#

show platform packet-trace packet 0

Packet: 0 CBUG ID: 120
Summary

Input : INJ.2

Output : GigabitEthernet0/0/1
State : FWD
Timestamp
Start : 115612780360228 ns (05/29/2014 15:02:55.467987 UTC)
Stop : 115612780380931 ns (05/29/2014 15:02:55.468008 UTC)
Path Trace
Feature: IPV4
Source : 172.16.10.1
Destination : 172.16.10.2
Protocol : 1 (ICMP)

```
ASR1000#
ASR1000#
```

```
show platform packet-trace packet 1
```

```
Packet: 1           CBUG ID: 121
Summary
Input     : GigabitEthernet0/0/1
Output    : internal0/0/rp:0
```

```
State     : PUNT 11  (For-us data)
```

```
Timestamp
Start   : 115612781060418 ns (05/29/2014 15:02:55.468687 UTC)
Stop    : 115612781120041 ns (05/29/2014 15:02:55.468747 UTC)
Path Trace
Feature: IPV4
Source      : 172.16.10.2
Destination : 172.16.10.1
Protocol    : 1 (ICMP)
```

Amélioration de Packet Trace avec IOSd et LFTS Punt/Inject Trace et correspondance UDF (nouveauté de la version 17.3.1)

La fonctionnalité de suivi des paquets est encore améliorée pour fournir des informations de suivi supplémentaires pour les paquets provenant ou destinés à IOSd ou à d'autres processus BinOS dans Cisco IOS-XE version 17.3.1.

IOSd Drop Tracing

Grâce à cette amélioration, le traçage de paquets est étendu dans IOSd, et peut fournir des informations sur les abandons de paquets à l'intérieur d'IOSd, qui sont généralement rapportés dans le résultat de la commande ***show ip traffic***. Aucune configuration supplémentaire n'est requise pour activer le suivi des abandons IOSd. Voici un exemple d’un paquet UDP abandonné par IOSd en raison d’une erreur de somme de contrôle incorrecte :

```
Router#debug platform condition ipv4 10.118.74.53/32 both
Router#debug platform condition start
Router#debug platform packet-trace packet 200
Packet count rounded up from 200 to 256

Router#
Router#show plat pack pa 0
Packet: 0           CBUG ID: 674
Summary
  Input     : GigabitEthernet1
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 17756544435656 ns (06/29/2020 18:19:17.326313 UTC)
    Stop    : 17756544469451 ns (06/29/2020 18:19:17.326346 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet1
    Output      : <unknown>
    Source      : 10.118.74.53
    Destination : 172.18.124.38
    Protocol    : 17 (UDP)
      SrcPort   : 2640
      DstPort   : 500

IOSd Path Flow: Packet: 0    CBUG ID: 674
  Feature: INFRA
  Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
  Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.118.74.53
    Destination : 172.18.124.38
    Interface   : GigabitEthernet1

  Feature: IP
  Pkt Direction: IN
  FORWARDED To transport layer
    Source        : 10.118.74.53
    Destination   : 172.18.124.38
    Interface     : GigabitEthernet1

  Feature: UDP
  Pkt Direction: IN
```

DROPPED

UDP: Checksum error: dropping

```
    Source      : 10.118.74.53(2640)
    Destination : 172.18.124.38(500)
```

Traçage du chemin de sortie IOSd

Le suivi des paquets est amélioré pour afficher les informations de suivi de chemin et de traitement de protocole lorsque le paquet provient de l'IOSd et est envoyé vers le réseau dans le sens de la sortie. Aucune configuration supplémentaire n'est requise pour capturer les informations de suivi du chemin de sortie IOSd. Voici un exemple de suivi du chemin de sortie pour un paquet SSH quittant le routeur :

```
      <#root>

Router#show platform packet-trace packet 2
Packet: 2           CBUG ID: 2

IOSd Path Flow:

  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 172.18.124.38:22 172.18.124.55:52774 seq 3052140910 OPTS 4 ACK 2346

  Feature: TCP
  Pkt Direction: OUT
  FORWARDED
    TCP: Connection is in SYNRCVD state
    ACK         : 2346709419
    SEQ         : 3052140910
    Source      : 172.18.124.38(22)
    Destination : 172.18.124.55(52774)


  Feature: IP
  Pkt Direction: OUTRoute out the generated packet.srcaddr: 172.18.124.38, dstaddr: 172.18.124.55

  Feature: IP
  Pkt Direction: OUTInject and forward successful srcaddr: 172.18.124.38, dstaddr: 172.18.124.55

  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 172.18.124.38:22 172.18.124.55:52774 seq 3052140910 OPTS 4 ACK 2346
Summary
  Input     : INJ.2
  Output    : GigabitEthernet1
  State     : FWD
  Timestamp
```

```
    Start   : 490928006866 ns (06/29/2020 13:31:30.807879 UTC)
    Stop    : 490928038567 ns (06/29/2020 13:31:30.807911 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : internal0/0/rp:0
    Output      : <unknown>
    Source      : 172.18.124.38
    Destination : 172.18.124.55
    Protocol    : 6 (TCP)
      SrcPort   : 22
      DstPort   : 52774
  Feature: IPSec
    Result    : IPSEC_RESULT_DENY
    Action    : SEND_CLEAR
    SA Handle : 0
    Peer Addr : 172.18.124.55
    Local Addr: 172.18.124.38
```

Suivi des paquets LFTS

LFTS (Linux Forwarding Transport Service) est un mécanisme de transport permettant de transférer des paquets envoyés par le CPP vers des applications autres que IOSd. L'amélioration du suivi des paquets LFTS a ajouté des informations de suivi pour ces paquets dans la sortie de suivi de chemin. Aucune configuration supplémentaire n'est requise pour obtenir les informations de traçage LFTS. Voici un exemple de sortie de traçage LFTS pour un paquet pointé vers l'application NETCONF :

<#root>

```
Router#show plat packet-trace pac 0
Packet: 0           CBUG ID: 461
Summary
  Input     : GigabitEthernet1
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 647999618975 ns (06/30/2020 02:18:06.752776 UTC)
    Stop    : 647999649168 ns (06/30/2020 02:18:06.752806 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet1
    Output      : <unknown>
    Source      : 10.118.74.53
    Destination : 172.18.124.38
    Protocol    : 6 (TCP)
      SrcPort   : 65365
      DstPort   : 830
```

LFTS Path Flow: Packet: 0   CBUG ID: 461

```
  Feature: LFTS
  Pkt Direction: IN
    Punt Cause  : 11
     subCause : 0
```

Correspondance du modèle de suivi des paquets basée sur le filtre défini par l'utilisateur (plate-forme ASR1000 uniquement)

Dans la version 17.3.1 de Cisco IOS-XE, un nouveau mécanisme de correspondance de paquets est également ajouté aux familles de produits ASR1000 pour correspondre à un champ arbitraire dans un paquet basé sur l'infrastructure de filtre défini par l'utilisateur (UDF). Cela permet une correspondance de paquets flexible basée sur des champs qui ne font pas partie de la structure d'en-tête L2/L3/L4 standard. L'exemple suivant montre une définition de FDU qui correspond à 2 octets du modèle défini par l'utilisateur 0x4D2 qui commence à partir d'un décalage de 26 octets de l'en-tête du protocole externe de couche 3.

```
udf grekey header outer l3 26 2
ip access-list extended match-grekey
 10 permit ip any any udf grekey 0x4D2 0xFFFF

debug plat condition ipv4 access-list match-grekey both
debug plat condition start
debug plat packet-trace pack 100
```

Exemples de Packet Trace

Cette section fournit quelques exemples où la fonctionnalité de suivi de paquets est utile à des fins de dépannage.

Exemple Packet Trace - NAT

Dans cet exemple, une traduction d'adresses réseau (NAT) source d'interface est configurée sur l'interface WAN d'un ASR1K (Gig0/0/0) pour le sous-réseau local (172.16.10.0/24).

Voici la configuration de la condition de plate-forme et de la trace de paquets qui est utilisée afin de suivre le trafic de 172.16.10.2 à 172.16.20.2, qui devient traduit (NAT) sur l'interface Gig0/0/0 :

```
debug platform condition interface Gig 0/0/1 ingress
debug platform condition start
```

```
debug platform packet-trace packet 1024 fia-trace
```

Lorsque cinq paquets ICMP sont envoyés de 172.16.10.2 à 172.16.20.2 avec une configuration NAT d’interface source, voici les résultats de la trace de paquets :

```
      <#root>

ASR1000#
```

show platform packet-trace summary

```
 Pkt   Input           Output          State  Reason
 0     Gi0/0/1         Gi0/0/0         FWD
 1     Gi0/0/1         Gi0/0/0         FWD
 2     Gi0/0/1         Gi0/0/0         FWD
 3     Gi0/0/1         Gi0/0/0         FWD
 4     Gi0/0/1         Gi0/0/0         FWD

ASR1000#
```

show platform packet-trace statistics

```
 Packets Summary
 Matched  5
 Traced   5
 Packets Received
 Ingress  5
 Inject   0
 Packets Processed
 Forward  5
 Punt     0
 Drop     0
 Consume  0

 ASR1000#
```

show platform packet-trace packet 0

```
 Packet: 0           CBUG ID: 146
 Summary
 Input     : GigabitEthernet0/0/1
 Output    : GigabitEthernet0/0/0
 State     : FWD
 Timestamp
 Start   : 3010217805313 ns (05/17/2014 07:01:52.227836 UTC)
 Stop    : 3010217892847 ns (05/17/2014 07:01:52.227923 UTC)
 Path Trace
 Feature: IPV4
 Source      : 172.16.10.2
 Destination : 172.16.20.2
 Protocol    : 1 (ICMP)
 Feature: FIA_TRACE
 Entry      : 0x806c7eac - DEBUG_COND_INPUT_PKT
 Lapsed time: 1031 ns
 Feature: FIA_TRACE
 Entry      : 0x82011c00 - IPV4_INPUT_DST_LOOKUP_CONSUME
 Lapsed time: 462 ns
 Feature: FIA_TRACE
 Entry      : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
 Lapsed time: 355 ns
 Feature: FIA_TRACE
 Entry      : 0x803c6af4 - IPV4_INPUT_VFR
 Lapsed time: 266 ns
 Feature: FIA_TRACE
 Entry      : 0x82004500 - IPV4_OUTPUT_LOOKUP_PROCESS
 Lapsed time: 942 ns
 Feature: FIA_TRACE
 Entry      : 0x8041771c - IPV4_INPUT_IPOPTIONS_PROCESS
 Lapsed time: 88 ns
 Feature: FIA_TRACE
 Entry      : 0x82013400 - MPLS_INPUT_GOTO_OUTPUT_FEATURE
 Lapsed time: 568 ns
 Feature: FIA_TRACE
 Entry      : 0x803c6900 - IPV4_OUTPUT_VFR
 Lapsed time: 266 ns
```

```
Feature: NAT
 Direction   : IN to OUT
 Action      : Translate Source
 Old Address : 172.16.10.2  00028
 New Address : 192.168.10.1 00002
```

```
 Feature: FIA_TRACE
 Entry      : 0x8031c248 - IPV4_NAT_OUTPUT_FIA
 Lapsed time: 55697 ns
 Feature: FIA_TRACE
 Entry      : 0x801424f8 - IPV4_OUTPUT_THREAT_DEFENSE
 Lapsed time: 693 ns
```

```
Feature: FIA_TRACE
Entry      : 0x803c60b8 - IPV4_MC_OUTPUT_VFR_REFRAG
Lapsed time: 88 ns
Feature: FIA_TRACE
Entry      : 0x82014900 - IPV6_INPUT_L2_REWRITE
Lapsed time: 444 ns
Feature: FIA_TRACE
Entry      : 0x82000080 - IPV4_OUTPUT_FRAG
Lapsed time: 88 ns
Feature: FIA_TRACE
Entry      : 0x8200e600 - IPV4_OUTPUT_DROP_POLICY
Lapsed time: 1457 ns
Feature: FIA_TRACE
Entry      : 0x82017980 - MARMOT_SPA_D_TRANSMIT_PKT
Lapsed time: 7431 ns
ASR1000#
```

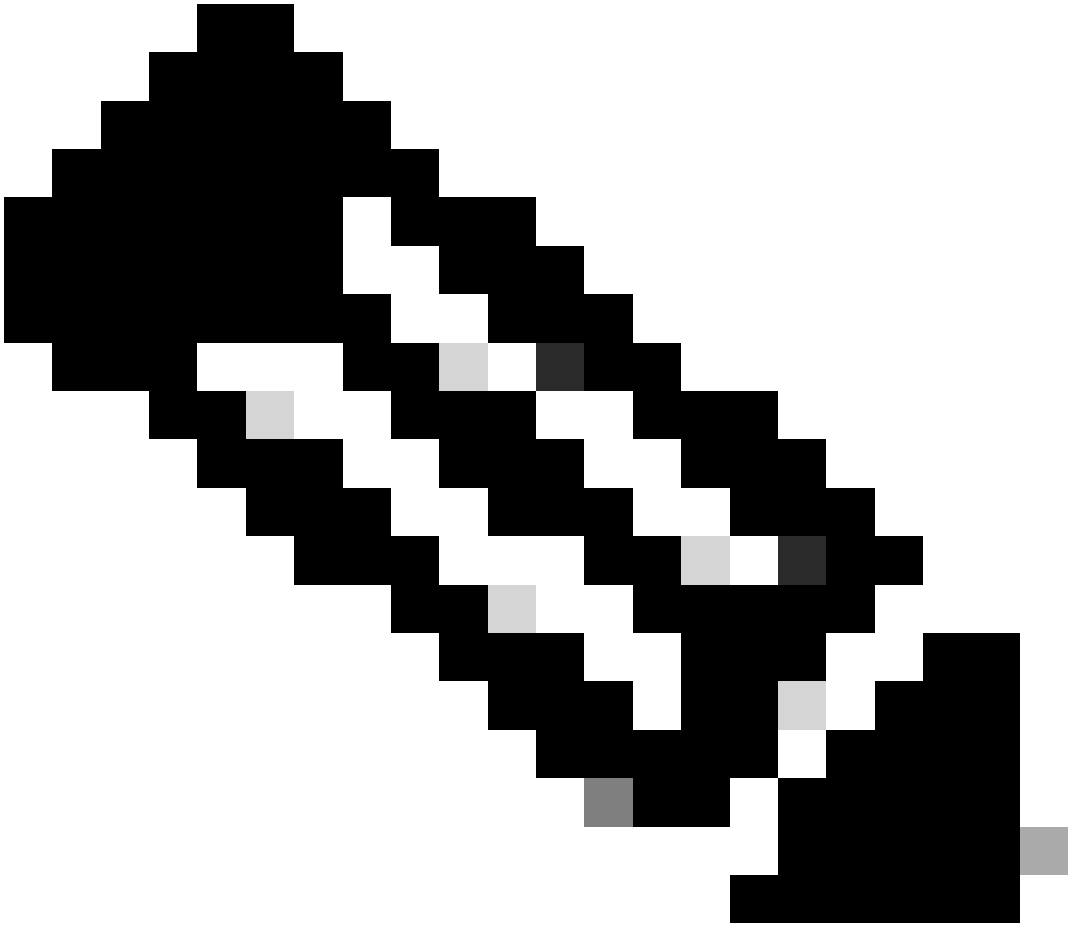
Exemple Packet Trace - VPN

Dans cet exemple, un tunnel VPN site à site est utilisé entre l'ASR1K et le routeur Cisco IOS afin de protéger le trafic qui circule entre 172.16.10.0/24 et 172.16.20.0/24 (sous-réseaux locaux et distants).

Voici la configuration de la condition de la plate-forme et du suivi des paquets qui est utilisée afin de suivre le trafic VPN qui circule de 172.16.10.2 à 172.16.20.2 sur l'interface Gig 0/0/1 :

```
debug platform condition interface Gig 0/0/1 ingress
debug platform condition start
debug platform packet-trace packet 1024 fia-trace
```

Lorsque cinq paquets ICMP sont envoyés de 172.16.10.2 à 172.16.20.2, qui sont chiffrés par le tunnel VPN entre l'ASR1K et le routeur Cisco IOS dans cet exemple, voici les sorties de trace de paquets :



Remarque : les suivis de paquets indiquent le handle de l'association de sécurité QFP dans le suivi utilisé pour chiffrer le paquet, ce qui est utile lorsque vous dépannez des problèmes VPN IPsec afin de vérifier que l'association de sécurité correcte est utilisée pour le chiffrement.

<#root>

ASR1000#

show platform packet-trace summary

```
Pkt   Input            Output           State  Reason
0     Gi0/0/1          Gi0/0/0          FWD
1     Gi0/0/1          Gi0/0/0          FWD
2     Gi0/0/1          Gi0/0/0          FWD
3     Gi0/0/1          Gi0/0/0          FWD
4     Gi0/0/1          Gi0/0/0          FWD

ASR1000#
```

show platform packet-trace packet 0

```
Packet: 0           CBUG ID: 211
Summary
Input     : GigabitEthernet0/0/1
Output    : GigabitEthernet0/0/0
State     : FWD
Timestamp
Start   : 4636921551459 ns (05/17/2014 07:28:59.211375 UTC)
Stop    : 4636921668739 ns (05/17/2014 07:28:59.211493 UTC)
Path Trace
Feature: IPV4
Source      : 172.16.10.2
Destination : 172.16.20.2
Protocol    : 1 (ICMP)
Feature: FIA_TRACE
Entry      : 0x806c7eac - DEBUG_COND_INPUT_PKT
Lapsed time: 622 ns
Feature: FIA_TRACE
Entry      : 0x82011c00 - IPV4_INPUT_DST_LOOKUP_CONSUME
Lapsed time: 462 ns
Feature: FIA_TRACE
Entry      : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
Lapsed time: 320 ns
Feature: FIA_TRACE
Entry      : 0x82004500 - IPV4_OUTPUT_LOOKUP_PROCESS
Lapsed time: 1102 ns
Feature: FIA_TRACE
Entry      : 0x8041771c - IPV4_INPUT_IPOPTIONS_PROCESS
Lapsed time: 88 ns
Feature: FIA_TRACE
Entry      : 0x82013400 - MPLS_INPUT_GOTO_OUTPUT_FEATURE
Lapsed time: 586 ns
Feature: FIA_TRACE
Entry      : 0x803c6900 - IPV4_OUTPUT_VFR
Lapsed time: 266 ns
Feature: FIA_TRACE
Entry      : 0x80757914 - MC_OUTPUT_GEN_RECYCLE
Lapsed time: 195 ns
Feature: FIA_TRACE
Entry      : 0x803c60b8 - IPV4_MC_OUTPUT_VFR_REFRAG
Lapsed time: 88 ns
```

```
Feature: IPSec
 Result    : IPSEC_RESULT_SA
 Action    : ENCRYPT
 SA Handle : 6
 Peer Addr : 192.168.20.1
 Local Addr: 192.168.10.1
```

```
 Feature: FIA_TRACE
 Entry      : 0x8043caec - IPV4_OUTPUT_IPSEC_CLASSIFY
 Lapsed time: 9528 ns
 Feature: FIA_TRACE
 Entry      : 0x8043915c - IPV4_OUTPUT_IPSEC_DOUBLE_ACL
 Lapsed time: 355 ns
 Feature: FIA_TRACE
 Entry      : 0x8043b45c - IPV4_IPSEC_FEATURE_RETURN
 Lapsed time: 657 ns
 Feature: FIA_TRACE
 Entry      : 0x8043ae28 - IPV4_OUTPUT_IPSEC_RERUN_JUMP
 Lapsed time: 888 ns
 Feature: FIA_TRACE
 Entry      : 0x80436f10 - IPV4_OUTPUT_IPSEC_POST_PROCESS
 Lapsed time: 2186 ns
 Feature: FIA_TRACE
 Entry      : 0x8043b45c - IPV4_IPSEC_FEATURE_RETURN
 Lapsed time: 675 ns
 Feature: FIA_TRACE
 Entry      : 0x82014900 - IPV6_INPUT_L2_REWRITE
 Lapsed time: 1902 ns
 Feature: FIA_TRACE
 Entry      : 0x82000080 - IPV4_OUTPUT_FRAG
 Lapsed time: 71 ns
 Feature: FIA_TRACE
 Entry      : 0x8200e600 - IPV4_OUTPUT_DROP_POLICY
 Lapsed time: 1582 ns
 Feature: FIA_TRACE
 Entry      : 0x82017980 - MARMOT_SPA_D_TRANSMIT_PKT
 Lapsed time: 3964 ns
 ASR1000#
```

Impact sur les performances

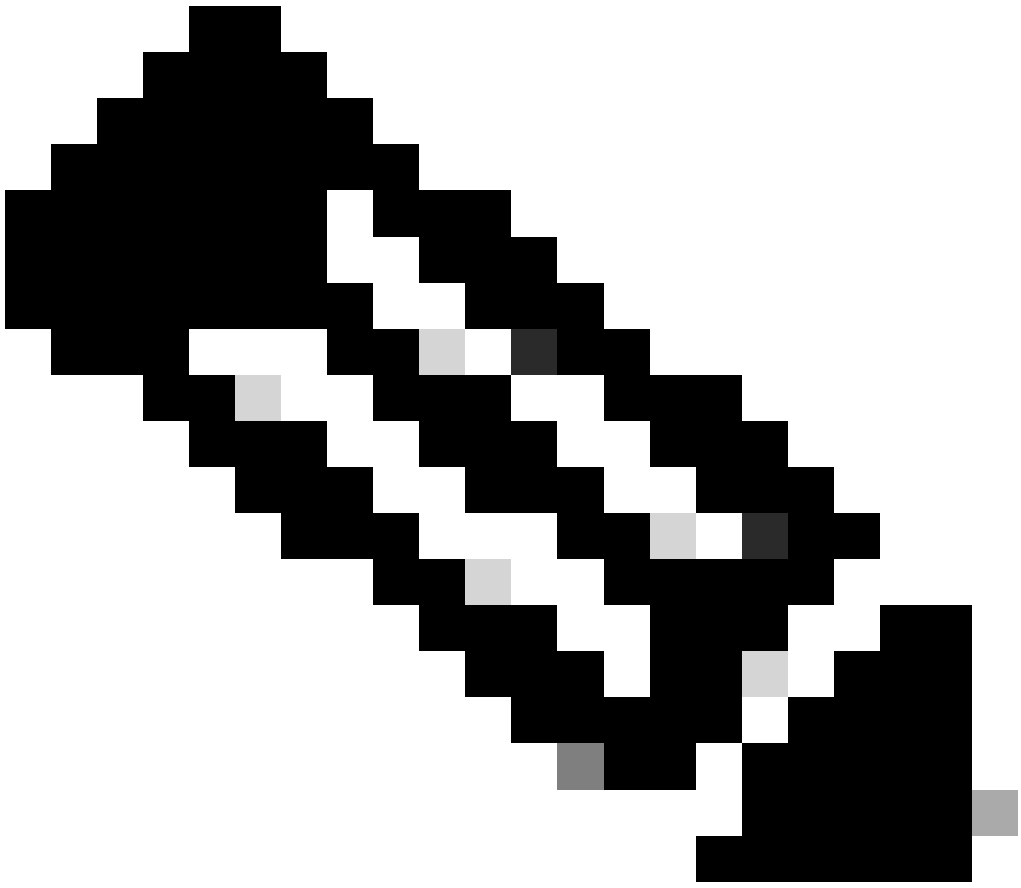
Les tampons de suivi de paquets consomment de la mémoire DRAM QFP. Par conséquent, n’oubliez pas la quantité de mémoire requise par une configuration et la quantité de mémoire disponible.

L'impact sur les performances varie en fonction des options de suivi des paquets activées. Le suivi des paquets affecte uniquement les performances de transfert des paquets suivis, tels que les paquets qui correspondent aux conditions configurées par l'utilisateur. Plus vous configurez le suivi des paquets pour capturer des informations précises et détaillées, plus l'impact sur les ressources est important.

Comme pour tout dépannage, il est préférable d'adopter une approche itérative et de n'activer les options de suivi plus détaillées que lorsqu'une situation de débogage le justifie.

L'utilisation de la mémoire DRAM QFP peut être estimée à l'aide de cette formule :

mémoire requise = (surcharge stats) + nombre de paquets * (taille de résumé + taille des données de chemin + taille de copie)



Remarque : lorsque la **surcharge d'état** et la **taille de résumé** sont fixées à 2 Ko et 128 Mo, respectivement, la **taille des données**

---

de chemin et la **taille de copie** sont configurables par l'utilisateur.

Informations connexes

- **Guide de configuration logicielle des routeurs de la gamme d'agrégation Cisco ASR1000 - Packet Trace**
- **Suppression de paquets sur les routeurs de service de la gamme Cisco ASR1000**
- **Assistance technique de Cisco et téléchargements**

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.