使用IOS-XE資料路徑資料包跟蹤功能排除故障

目錄

簡介

本檔案介紹如何透過封包追蹤功能執行Cisco IOS-XE®軟體的資料路徑封包追蹤。

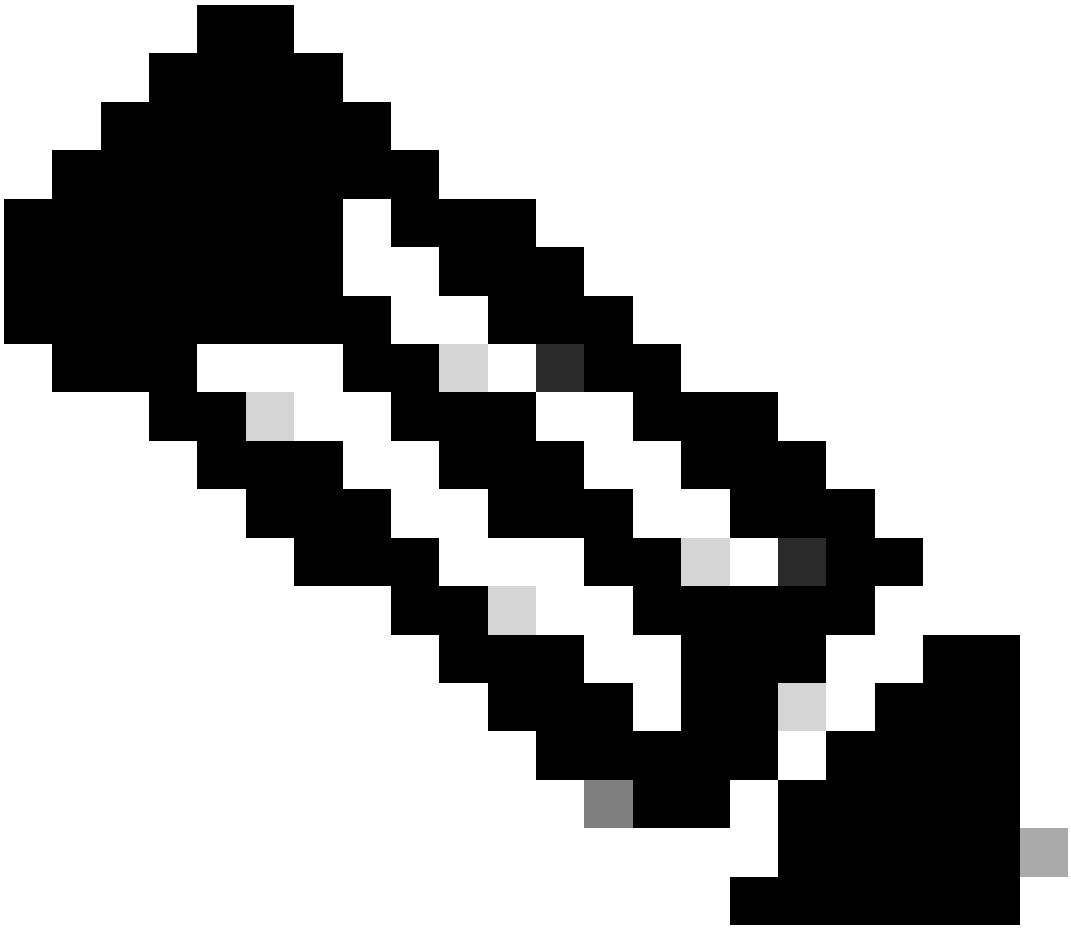
必要條件

需求

思科建議您瞭解以下資訊：

在基於QFP（量子流處理器）的路由平台（包括ASR1000、ISR4000、ISR1000、Catalyst 1000、

Catalyst 8000、CSR1000v和Catalyst 8000v系列路由器）上的Cisco IOS-XE 3.10版和更高版本中提供了資料包跟蹤功能。運行Cisco IOS-XE軟體的ASR900系列聚合服務路由器或Catalyst系列交換機不支援此功能。



注意：資料包跟蹤功能在ASR1000系列路由器上的專用管理介面GigabitEthernet0上不起作用，因為在該介面上轉發的資料包不由QFP處理。

採用元件

本文中的資訊係根據以下軟體和硬體版本：

- Cisco IOS-XE軟體版本3.10S (15.3(3)S)及更高版本
- ASR1000系列路由器

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

背景資訊

若要在疑難排解時找出組態錯誤、容量超載，甚至一般的軟體錯誤等問題，必須瞭解系統內封包會發生什麼情況。Cisco IOS-XE資料包跟蹤功能可以滿足這一需求。它提供了一種欄位安全方法，用於記帳，並根據使用者定義條件類捕獲每個資料包的進程詳細資訊。

參考拓撲

此圖說明用於本文檔所述示例的拓撲：

使用中的資料包跟蹤

為了說明封包追蹤功能的使用，本節使用的範例說明在ASR1K的介面GigabitEthernet0/0/1上，從本機工作站172.16.10.2（位於ASR1K之後）到遠端主機172.16.20.2的網際網路控制訊息通訊協定(ICMP)流量在輸入方向的追蹤。

您可以使用以下兩個步驟跟蹤ASR1K上的資料包：

1. 啟用平台條件調試，以選擇要在ASR1K上跟蹤的資料包或流量。

2. 使用path-trace或Feature Invocation Array (FIA)跟蹤選項啟用平台資料包跟蹤。

快速入門手冊

如果您已經熟悉本文檔的內容，並且希望使用一節來快速檢視CLI，請閱讀本快速入門手冊。以下僅是說明工具使用的一些範例。請參閱後面幾節，詳細討論語法，並確保使用符合您要求的配置。

1. 配置平台條件：

```
<#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)
```

配置平台條件後，使用以下CLI命令啟動平台條件：

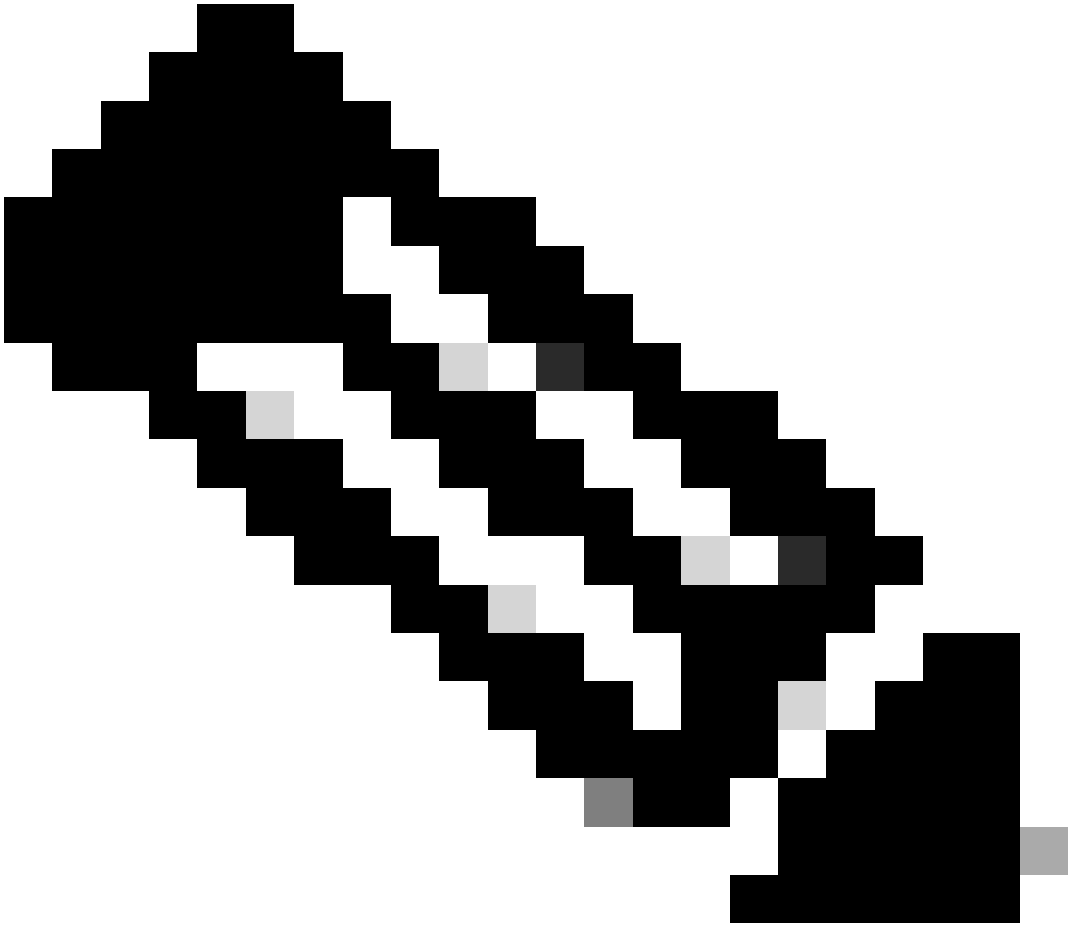
```
<#root>

debug platform condition start
```

2. 配置資料包跟蹤：

```
<#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.
```



注意：在早期的Cisco IOS-XE 3.x版本中，還需要使用debug platform packet-trace enable命令才能啟動資料包跟蹤功能。Cisco IOS-XE 16.x版本不再需要此功能。

輸入以下命令可清除追蹤緩衝區和reset packet-trace：

<#root>

```
clear platform packet-trace statistics
```

```
  --> clear the packet trace buffer
```

用於清除平台條件和資料包跟蹤配置的命令為：

<#root>

```
clear platform condition all
```

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

顯示命令

應用上述命令後，請驗證平台條件和資料包跟蹤配置，以確保獲得所需的配置。

<#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
```

以下是用於檢查追蹤/擷取封包的命令：

<#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
```

啟用平台條件式偵錯

封包追蹤功能依賴條件式偵錯基礎架構來確定要追蹤的封包。條件式偵錯基礎架構提供根據以下條件過濾流量的功能：

- 通訊協定
- IP地址和掩碼
- 存取控制清單(ACL)

- 介面
- 流量方向（入口或出口）

這些條件定義了過濾器應用於資料包的位置和時間。

對於本示例中使用的流量，為從172.16.10.2到172.16.20.2的ICMP資料包啟用入口方向上的平台條件調試。換句話說，選擇要跟蹤的流量。可選擇此流量可使用各種選項。

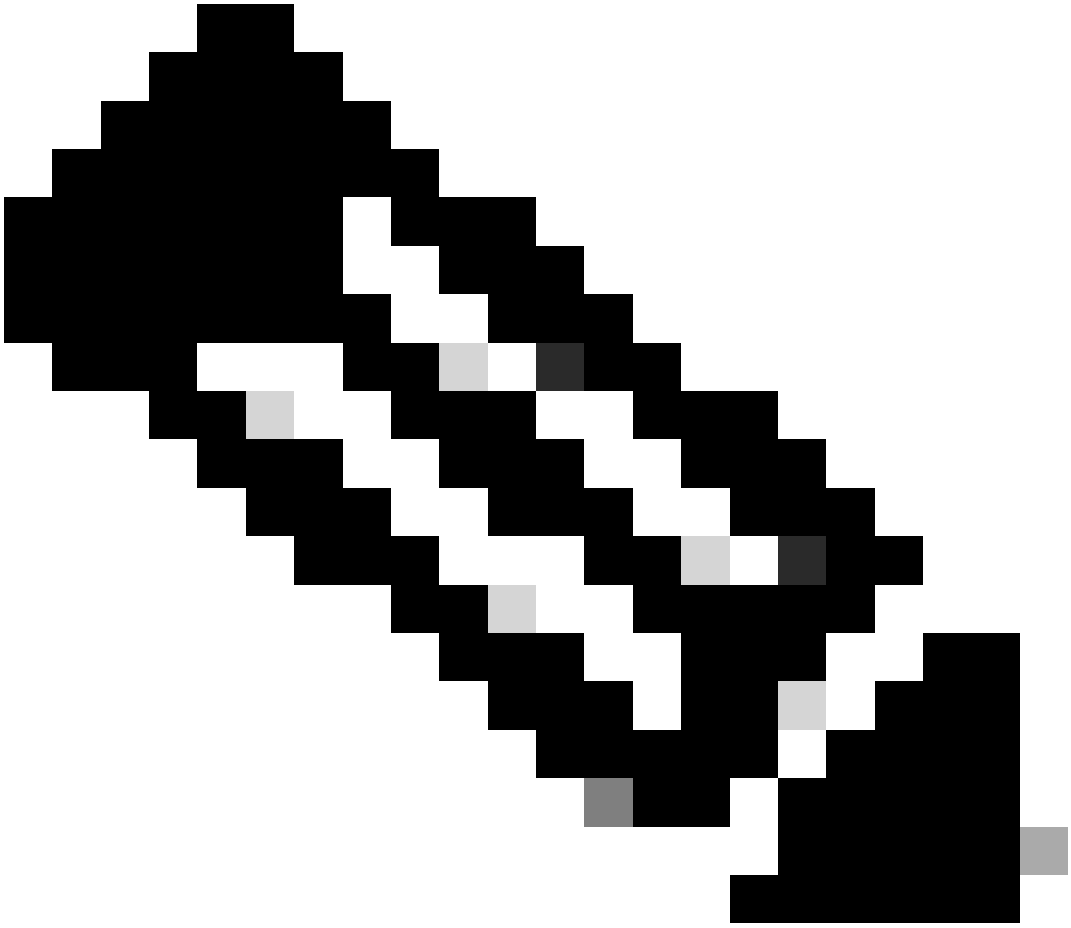
```
<#root>

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
```

在此範例中，會使用存取清單來定義條件，如下所示：

```
<#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
```

若要開始條件式除錯，請輸入以下命令：

```
<#root>

ASR1000#

debug platform condition start
```



注意：要停止或停用條件調試基礎結構，請輸入debug platform condition stop命令。

若要檢視已設定的條件式偵錯篩選條件，請輸入以下命令：

<#root>

ASR1000#

show platform conditions

Conditional Debug Global State:

Start

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

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

ASR1000#
```

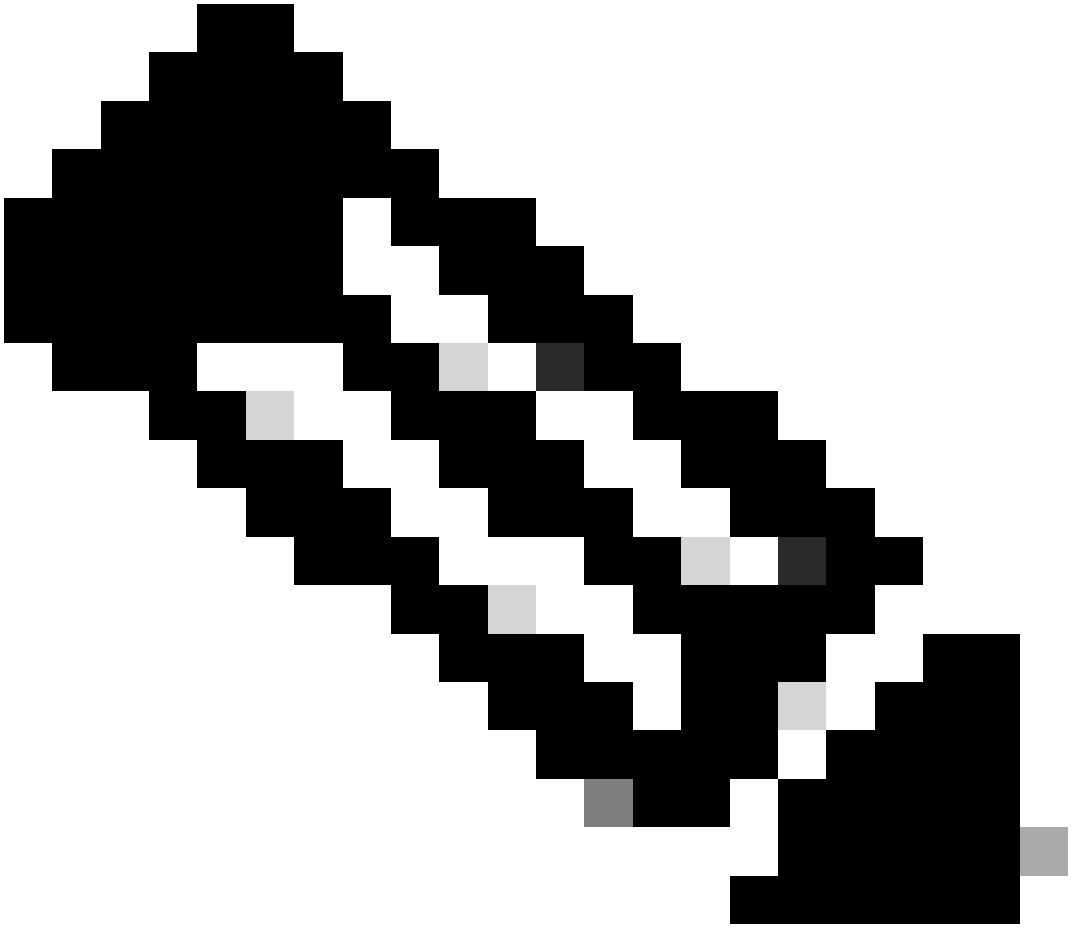
總而言之，到目前為止已應用此配置：

<#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

啟用資料包跟蹤



注意：本部分詳細介紹資料包和複製選項，其他選項將在本文檔後面部分介紹。

物理介面和邏輯介面（例如隧道介面或虛擬訪問介面）都支援資料包跟蹤。

以下是封包追蹤CLI語法：

<#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>]

以下是此命令關鍵字的說明：

- pkt-num - Packet Number指定同時維護的資料包的最大數量。
- summary-only -用於指定僅捕獲摘要資料。預設情況下會同時捕獲摘要資料和功能路徑資料。
- fia-trace -除了路徑資料資訊外，它還選擇性地執行FIA跟蹤。
- data-size -可用於指定路徑資料緩衝區的大小，從2,048到16,384位元組。預設值為2,048位元組。

<#root>

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

以下是此命令關鍵字的說明：

- in/out -用於指定要複製的資料包流的方向-入口和/或出口。
- L2/L3/L4 -可用於指定資料包副本的起始位置。第2層(L2)是預設位置。

- size -可讓您指定複製的八位元最大數目。預設值為64個八位組。

在本範例中，以下命令用於為使用條件式偵錯基礎架構選取的流量啟用封包追蹤：

```
<#root>
ASR1000#
debug platform packet-trace packet 16
```

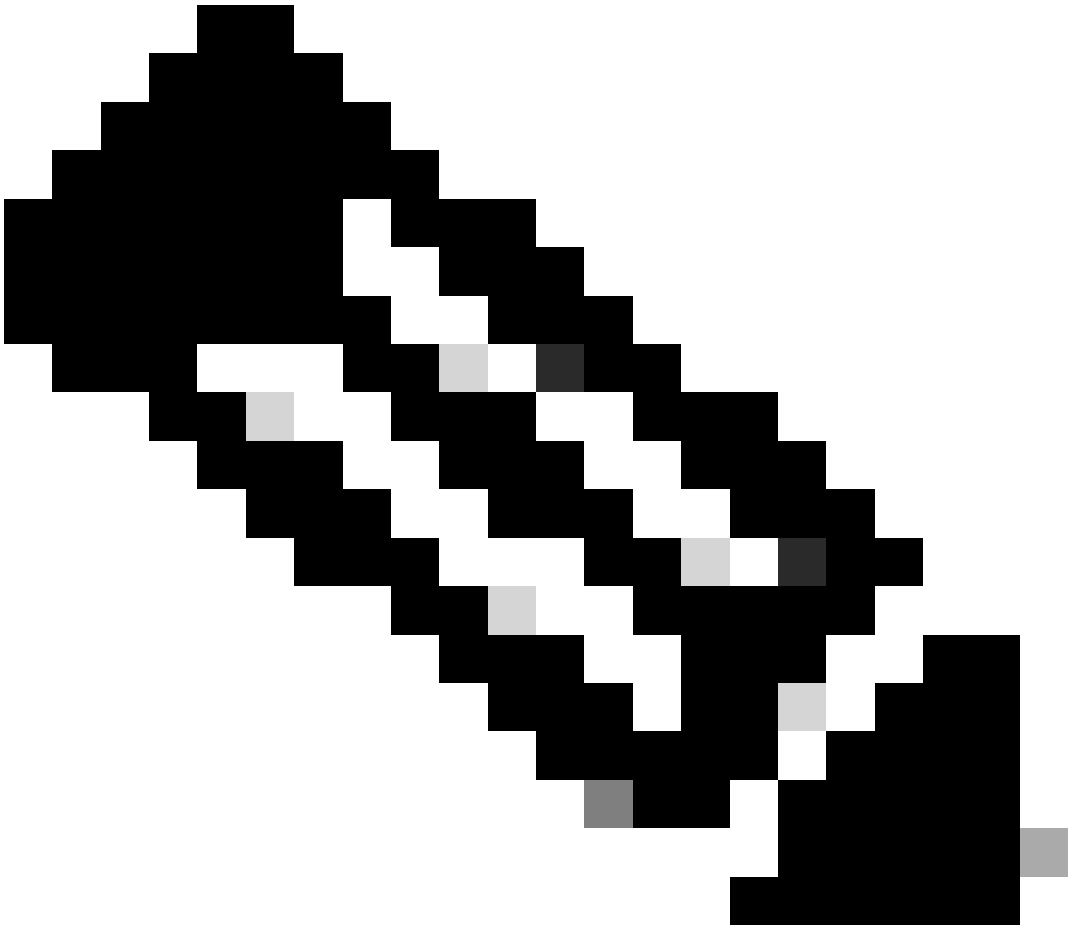
若要檢視封包追蹤組態，請輸入以下命令：

```
<#root>
ASR1000#
show platform packet-trace configuration

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

還可以輸入show debugging命令檢視平台條件調試和資料包跟蹤配置：

```
<#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
```



注意：輸入clear platform condition all命令以清除所有平台調試條件和資料包跟蹤配置和資料。

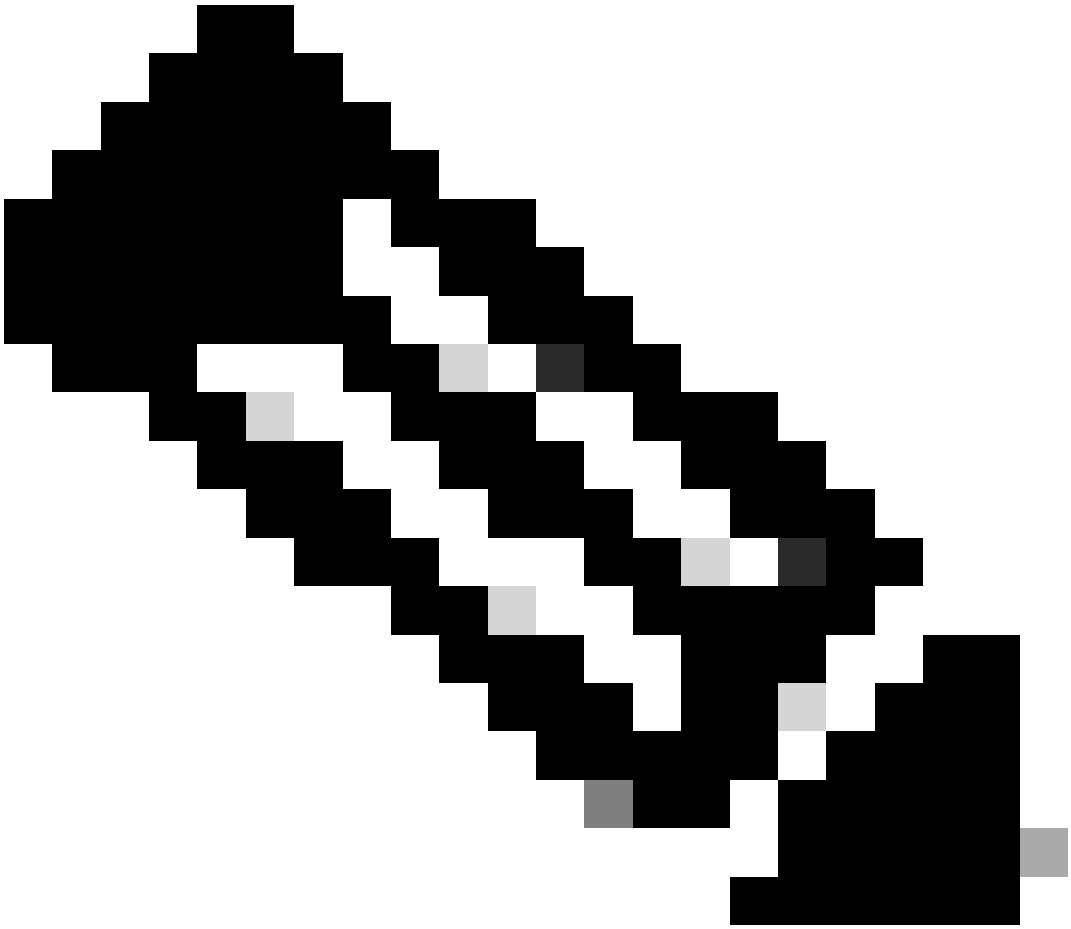
總而言之，到目前為止使用此配置資料來啟用資料包跟蹤：

<#root>

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

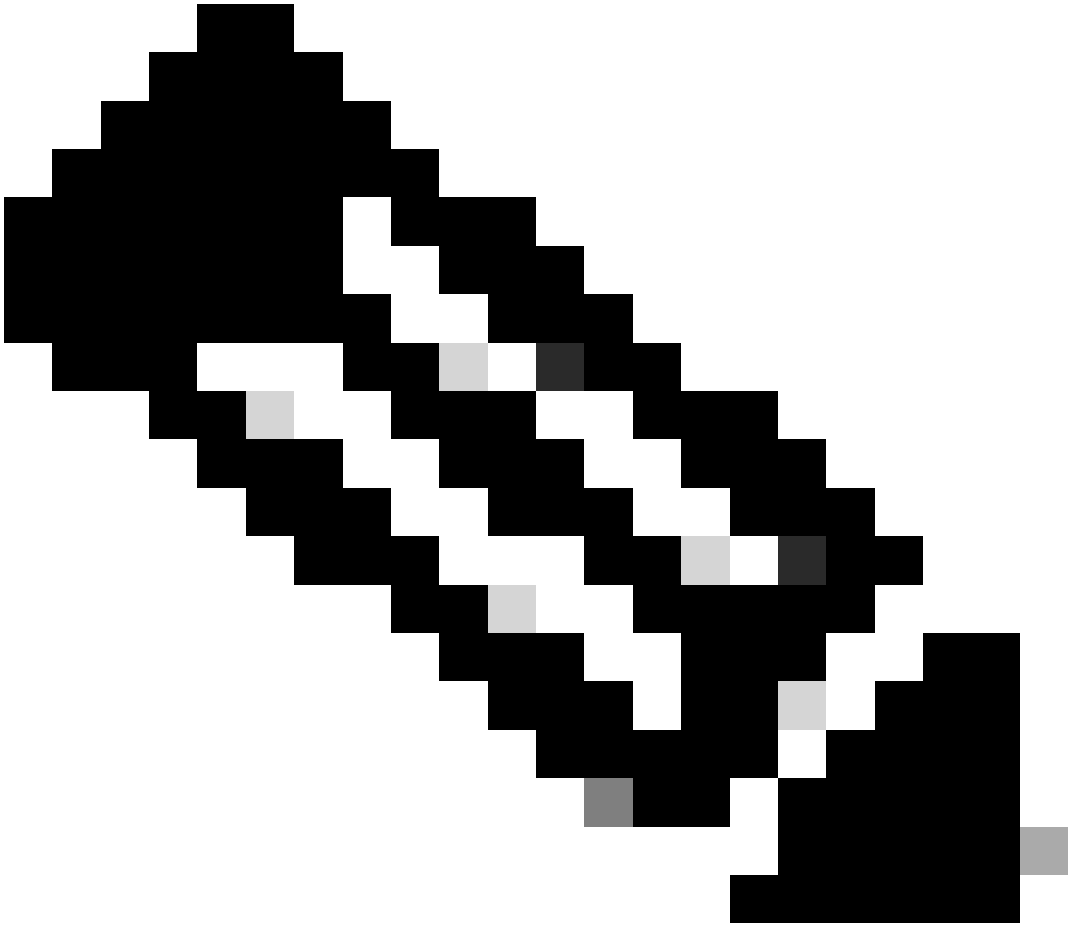
封包追蹤的出口條件限制

這些條件定義了條件過濾器以及將其應用於資料包的時間。例如，debug platform condition interface g0/0/0 egress意味著當資料包到達介面g0/0/0上的輸出FIA時，會將其標識為匹配，因此從入口到該點的所有資料包處理都將丟失。



注意：思科強烈建議您對資料包跟蹤使用入口條件，以便獲得儘可能完整和有意義的資料。可以使用出口條件，但請注意其限制。

顯示資料包跟蹤結果



注意：本部分假定路徑跟蹤已啟用。

資料包跟蹤提供三個特定級別的檢查：

- 計量
- 每個資料包的摘要
- 每資料包路徑資料

從172.16.10.2到172.16.20.2傳送五個ICMP請求資料包時，可以使用以下命令檢視資料包跟蹤結果：

<#root>

ASR1000#

show platform packet-trace statistics

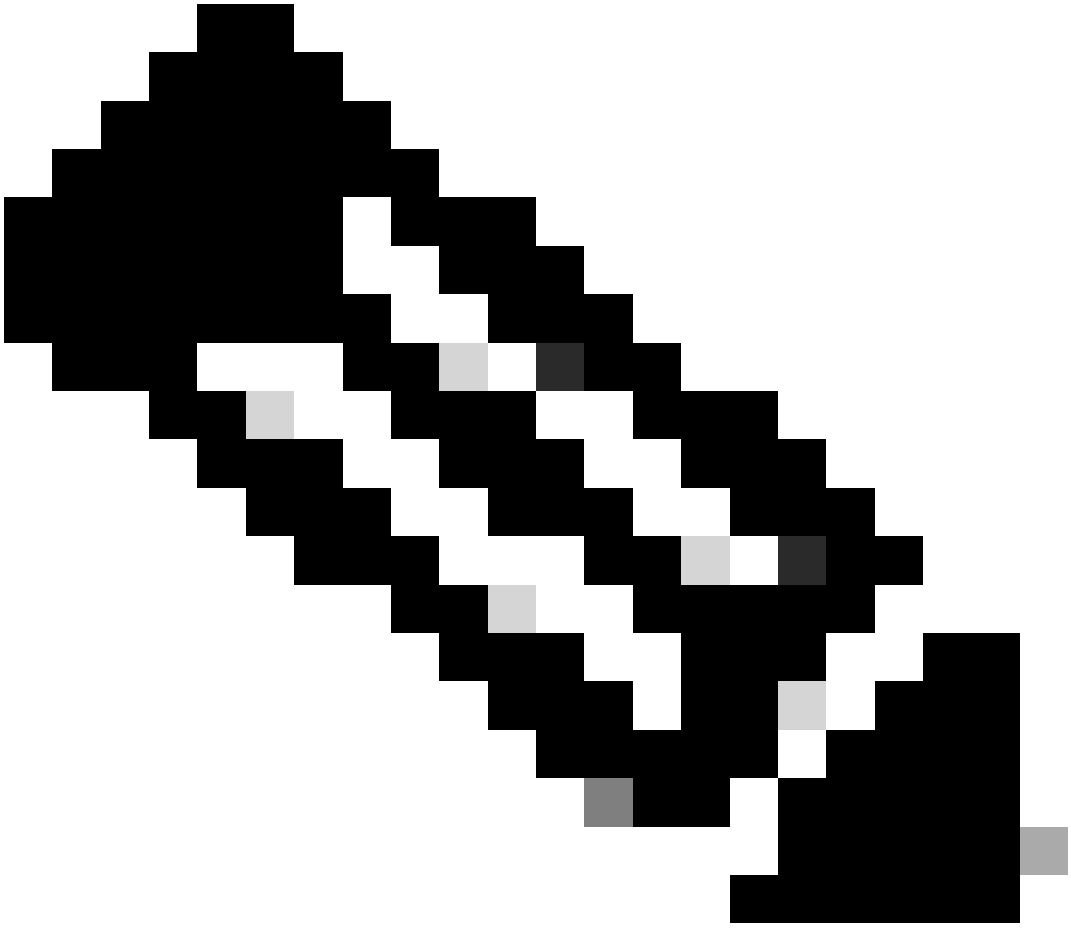
```
Packets Traced: 5


 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#
```



注意：第三個命令提供了一個示例，演示如何檢視每個資料包的資料包跟蹤。本例中顯示第一個跟蹤的資料包。

從這些輸出中，您可以看到跟蹤了五個資料包，並且您可以檢視輸入介面、輸出介面、狀態和路徑跟蹤。

狀態	備註
前驅	資料包已排程/排隊等待傳輸，並透過出口介面轉發到下一跳。
棄置	封包從轉送處理器(FP)傳送到路由處理器(RP)（控制平面）。
DROP	資料包在FP上被丟棄。運行FIA跟蹤、使用全局丟棄計數器或使用資料路徑調試來查詢有關丟棄原因的更多詳細資訊。
缺點	封包會在封包處理過程中（例如ICMP Ping要求或加密封包）使用。

在資料包跟蹤統計資訊輸出中，ingress和inject計數器分別對應於透過外部介面進入的資料包和被視為從控制平面注入的資料包。

FIA追蹤

FIA包含當資料包轉發到入口或出口時，Quantum Flow處理器(QFP)中的資料包處理器引擎(PPE)按順序執行的功能清單。這些功能基於電腦上應用的配置資料。因此，FIA跟蹤有助於瞭解資料包在處理過程中透過系統的流量。

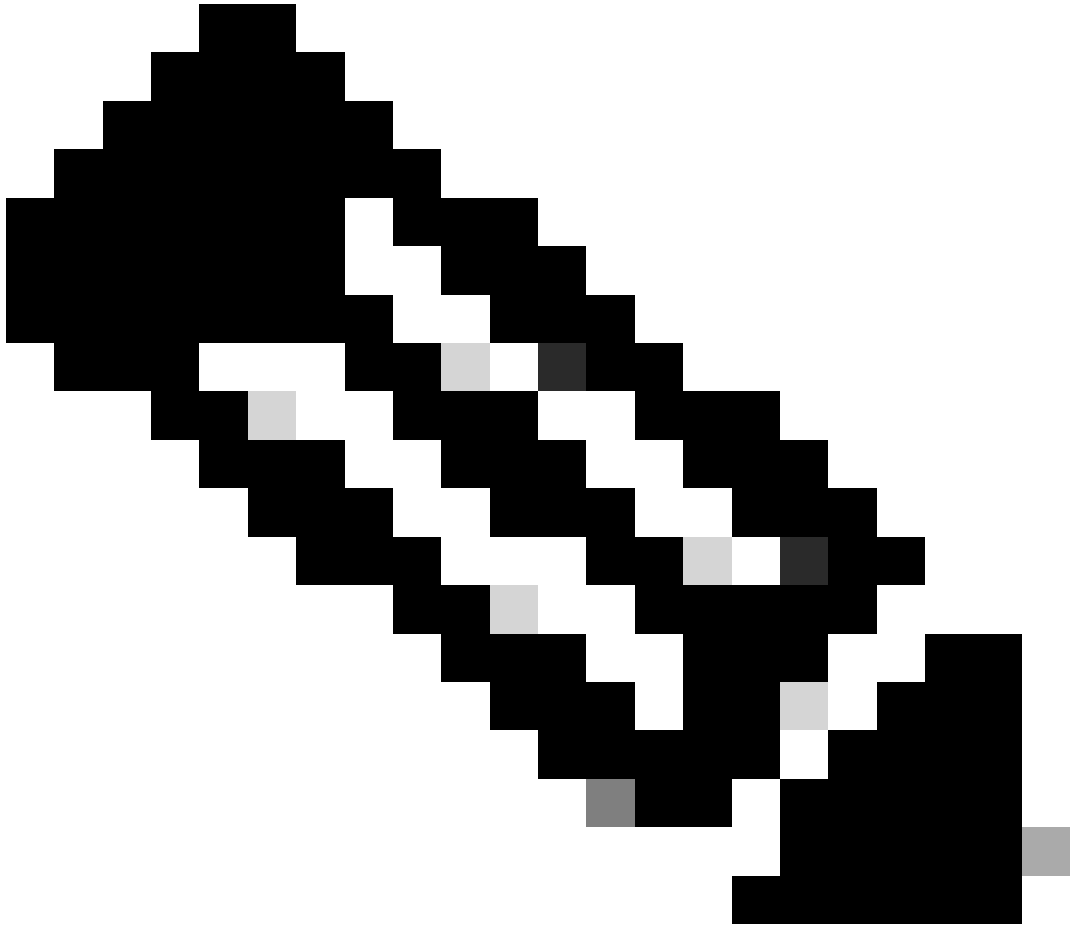
您必須應用此配置資料，才能使用FIA啟用資料包跟蹤：

<#root>

ASR1000#

debug platform packet-trace packet 16 fia-trace

顯示資料包跟蹤結果



注意：本部分假定已啟用FIA跟蹤。此外，當您增加或修改當前資料包跟蹤命令時，緩衝的

資料包跟蹤詳細資訊會被清除，因此您必須再次傳送一些流量才能對其進行跟蹤。

在輸入用於啟用FIA跟蹤的命令後，從172.16.10.2向172.16.20.2傳送五個ICMP資料包，如上一節所述。

```
<#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#
```

檢查與介面關聯的FIA

當您啟用平台條件調試時，條件調試會作為功能增加到FIA。根據介面上處理的功能順序，需要相應地設定條件過濾器，例如，在條件過濾器中必須使用NAT地址的前後地址。

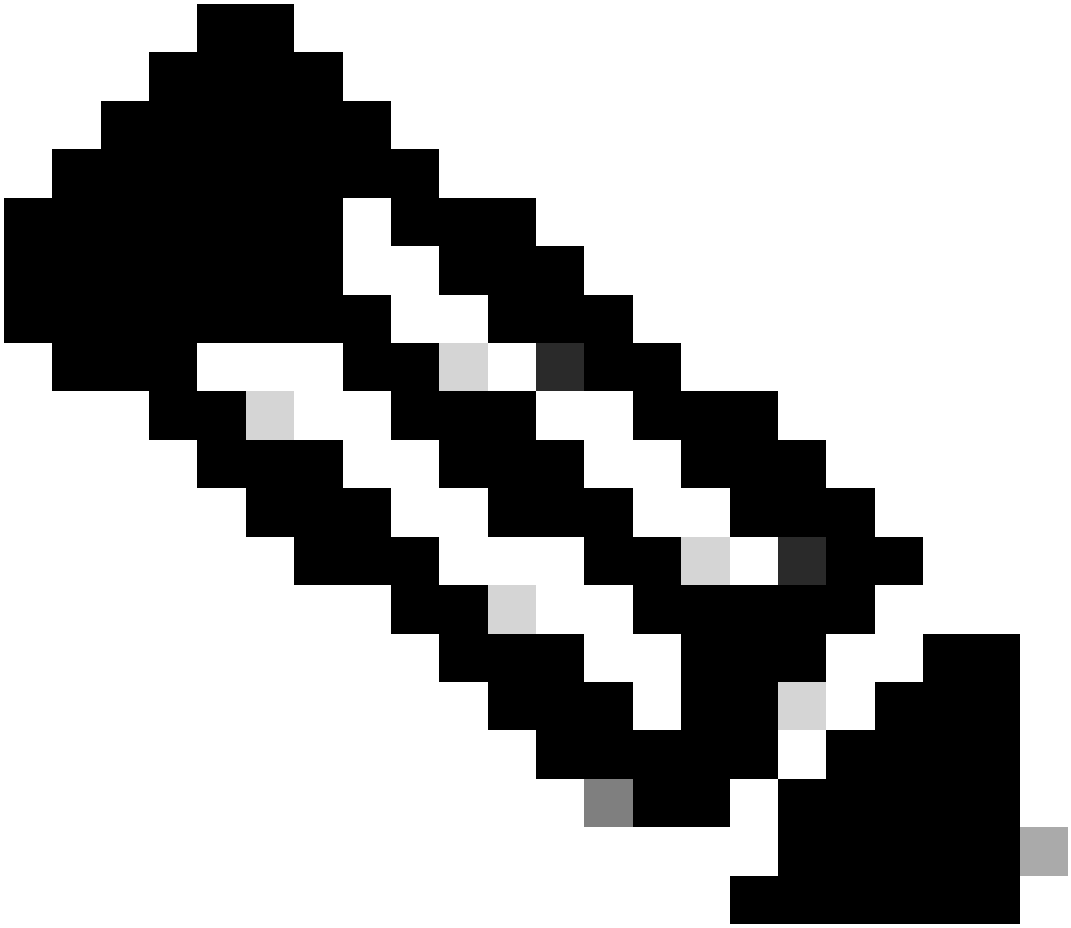
此輸出顯示在入口方向上啟用的平台條件調試的FIA中的功能順序：

<#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#
```



注意： CBUG_INPUT_FIA和DEBUG_COND_INPUT_PKT對應於路由器上配置的條件調試功能。

傾印追蹤的封包

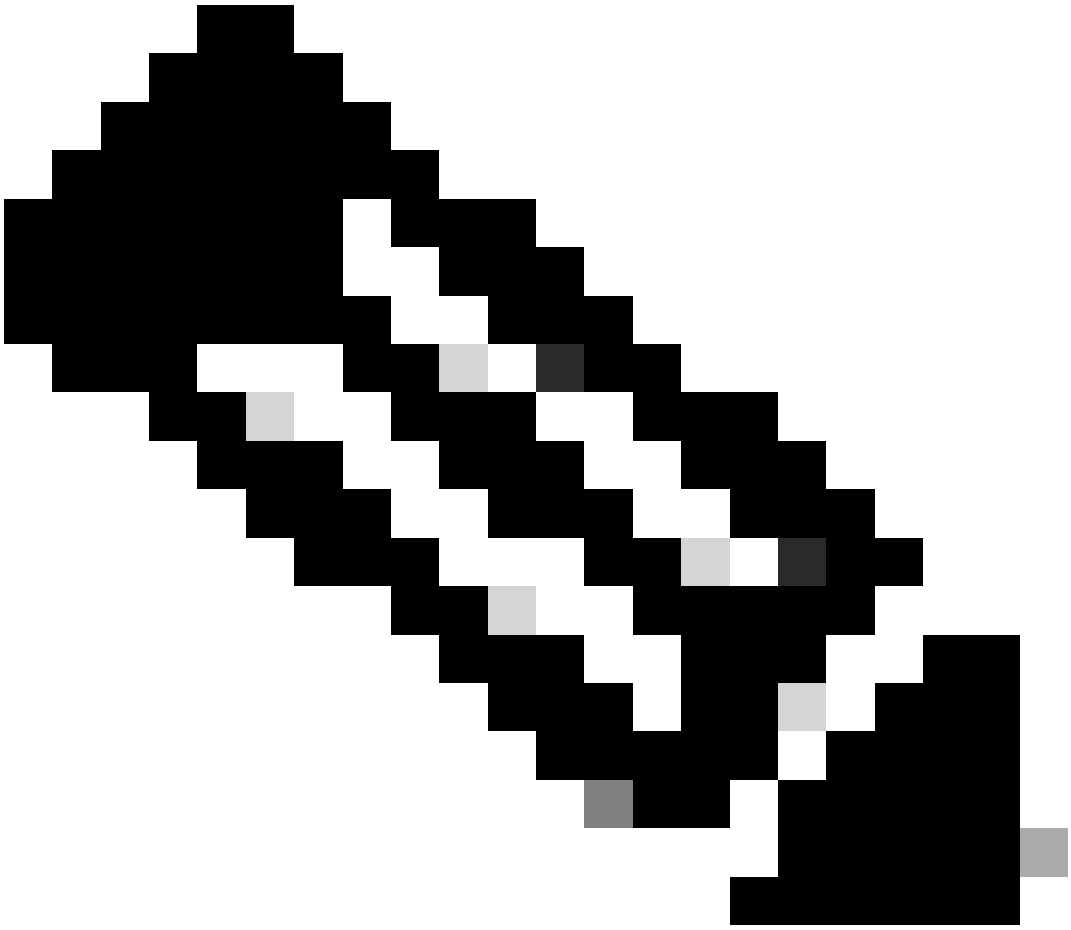
您可以在跟蹤資料包時複製和轉儲這些資料包，如本節所述。此範例顯示如何在輸入方向（172.16.10.2到172.16.20.2）複製最多2,048位元組的封包。

以下是需要的額外命令：

<#root>

ASR1000#

debug platform packet-trace copy packet input size 2048



注意：複製的資料包大小在16到2,048位元組的範圍內。

輸入以下命令可轉儲複製的資料包：

<#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#
```

刪除跟蹤

Cisco IOS-XE軟體版本3.11及更高版本中提供了丟棄跟蹤。它僅對丟棄的資料包啟用資料包跟蹤。以下是功能的一些亮點：

- 它選擇性地允許您為特定丟棄代碼指定資料包的保留。
- 它可在沒有全局或介面條件的情況下用於捕獲丟棄事件。
- 丟棄事件捕獲意味著僅跟蹤丟棄本身，而不跟蹤資料包的壽命。但是，仍允許您捕獲摘要資料、元組資料和資料包，以便幫助細化條件或提供下一個調試步驟的線索。

以下是用於啟用捨棄型別封包追蹤的命令語法：

<#root>

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

丟棄代碼與丟棄ID相同，如show platform hardware qfp active statistics drop detail命令輸出中所報告的：

<#root>

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

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

刪除追蹤案例範例

將以下ACL應用於ASR1K的Gig 0/0/0介面，以丟棄從172.16.10.2到172.16.20.2的流量：

```
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
```

ACL會捨棄從本機主機到遠端主機的流量，因此請套用此捨棄追蹤組態：

<#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
```

將五個ICMP請求資料包從172.16.10.2傳送到172.16.20.2。丟棄跟蹤可捕獲到ACL丟棄的這些資料包，如下所示：

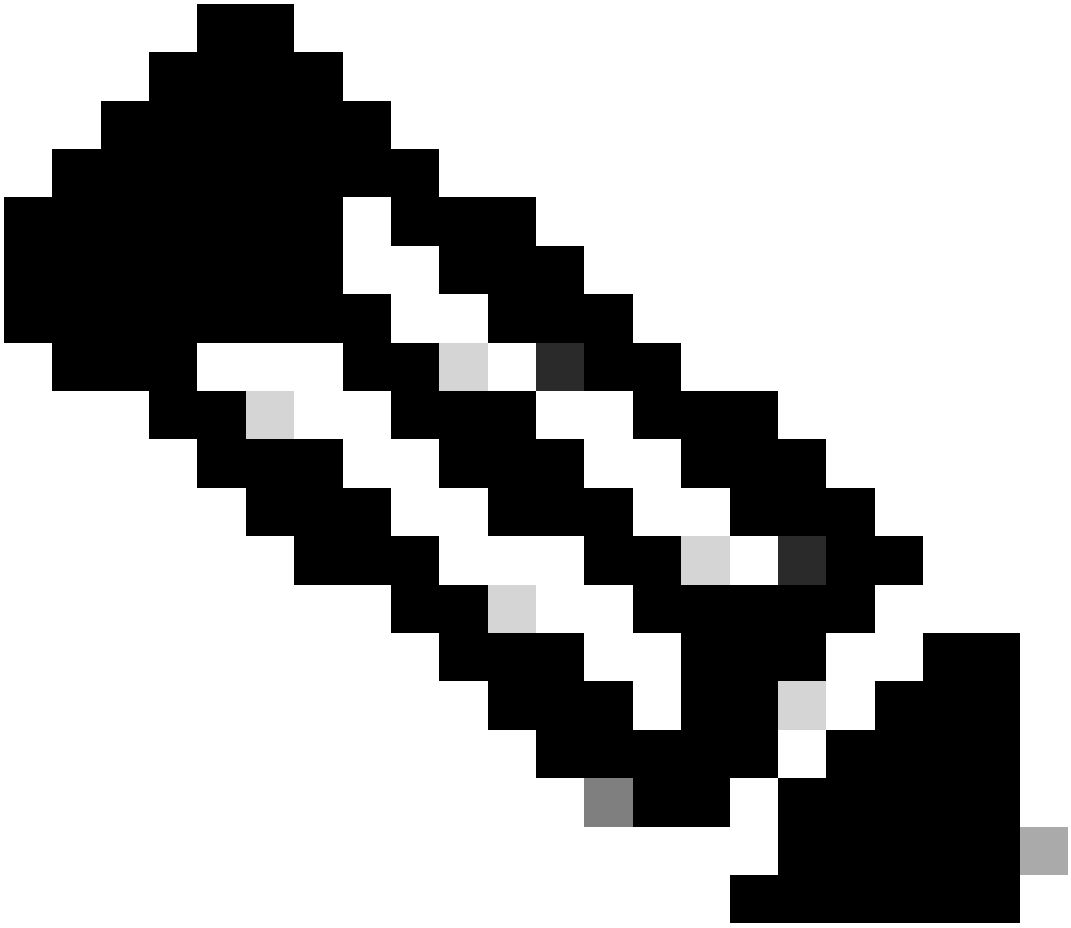
<#root>

```
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#
```

插入和穿刺痕跡

在Cisco IOS-XE軟體版本3.12及更高版本中增加了插入和傳送資料包跟蹤功能，以便跟蹤傳送（在FP上接收的被傳送至控制平面的資料包）和插入（從控制平面被插入到FP的資料包）資料包。



注意：棄置跟蹤可以在沒有全局或介面條件的情況下工作，就像丟棄跟蹤一樣。但是，必須定義條件才能使注入跟蹤正常工作。

以下是從ASR1K ping相鄰路由器時punt 和inject packet trace 的示例：

<#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#
```

現在，您可以驗證punt 和nject trace riesults：

<#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)
```

使用IOSd和LFTS傳送/插入跟蹤和UDF匹配增強資料包跟蹤（17.3.1中的新功能）

在Cisco IOS-XE版本17.3.1中，封包追蹤功能進一步增強，以便為來源或目的地為IOSd或其他BinOS處理序的封包提供額外的追蹤資訊。

IOSd丟棄跟蹤

透過此增強功能，資料包跟蹤擴展到IOSd，並且可以提供有關IOSd內部任何資料包丟棄的資訊，這些資訊通常在***show ip traffic***輸出中報告。啟用IOSd丟棄跟蹤不需要其他配置。以下範例顯示由於錯誤總和檢查碼錯誤而IOSd捨棄的UDP封包：

```
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)
```

IOSd輸出路徑追蹤

資料包跟蹤功能得到增強，可顯示路徑跟蹤和協定處理資訊，因為資料包是從IOSd發起的，並且以出口方向傳送到網路。捕獲IOSd出口路徑跟蹤資訊不需要其他配置。以下是路由器傳出的SSH資料包的出口路徑跟蹤示例：

```
<#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
```

LFTS資料包跟蹤

LFTS（Linux轉發傳輸服務）是一種傳輸機制，用於將從CPP傳送的資料包轉發到IOSd以外的應用程式中。LFTS資料包跟蹤增強功能在路徑跟蹤輸出中增加了此類資料包的跟蹤資訊。獲取LFTS跟蹤資訊不需要其他配置。以下是傳送至NETCONF應用程式的封包之LFTS追蹤的輸出範例：

<#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
```

基於使用者定義的過濾器的資料包跟蹤模式匹配（僅限ASR1000平台）

在Cisco IOS-XE版本17.3.1中，ASR1000產品系列中還增加了一個新的資料包匹配機制，用於根據使用者定義的過濾器(UDF)基礎設施匹配資料包中的任意欄位。這允許根據不是標準L2/L3/L4報頭結構一部分的欄位進行靈活的資料包匹配。下一個範例顯示的UDF定義符合2位元組的0x4D2使用者定義模式，該模式從與L3外部通訊協定標頭的26位元組位移開始。

```
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
```

資料包跟蹤示例

本節提供一些資料包跟蹤功能可用於故障排除的示例。

資料包跟蹤示例- NAT

在本示例中，在本地子網(172.16.10.0/24)的ASR1K (Gig0/0/0)的WAN介面上配置介面源網路地址轉換(NAT)。

以下是用於跟蹤從172.16.10.2到172.16.20.2（在Gig0/0/0介面上變為轉換[NAT]）的流量的平台條件和資料包跟蹤配置：

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

當使用介面源NAT配置從172.16.10.2傳送到172.16.20.2的五個ICMP資料包時，以下是資料包跟蹤結果：

<#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#
```

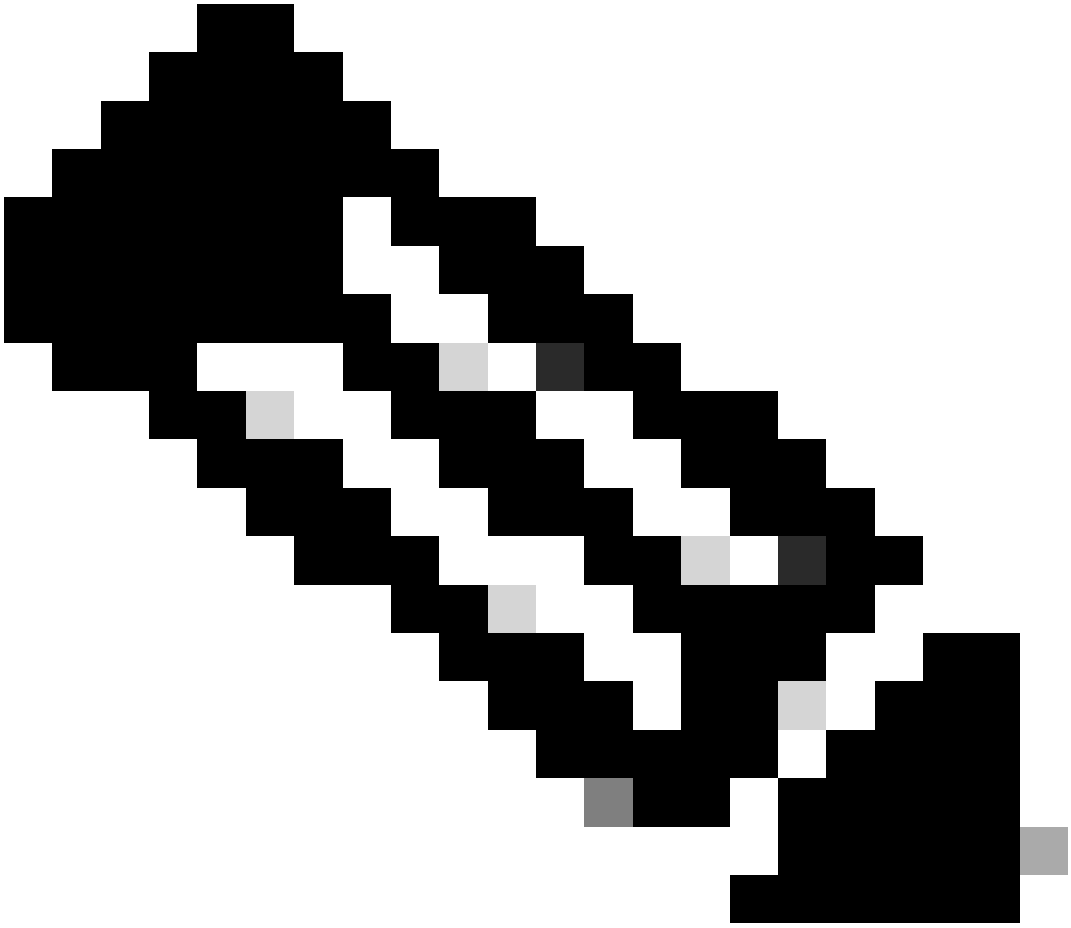
資料包跟蹤示例- VPN

在本示例中，在ASR1K和Cisco IOS路由器之間使用站點到站點VPN隧道，以保護在172.16.10.0/24和172.16.20.0/24（本地和遠端子網）之間流動的流量。

以下是用於跟蹤在Gig 0/0/1介面上從172.16.10.2流向172.16.20.2的VPN流量的平台條件和資料包跟蹤配置：

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

在本例中，從172.16.10.2傳送到172.16.20.2的五個ICMP資料包透過ASR1K與Cisco IOS路由器之間的VPN隧道進行加密時，資料包跟蹤輸出如下：



注意：資料包跟蹤顯示用於加密資料包的跟蹤中的QFP安全關聯(SA)控制代碼，當您排除IPsec VPN故障以驗證是否使用了正確的SA進行加密時，該控制代碼非常有用。

<#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#
```

效能影響

封包追蹤緩衝區會耗用QFP DRAM，因此請注意組態所需的記憶體數量以及可用的記憶體數量。

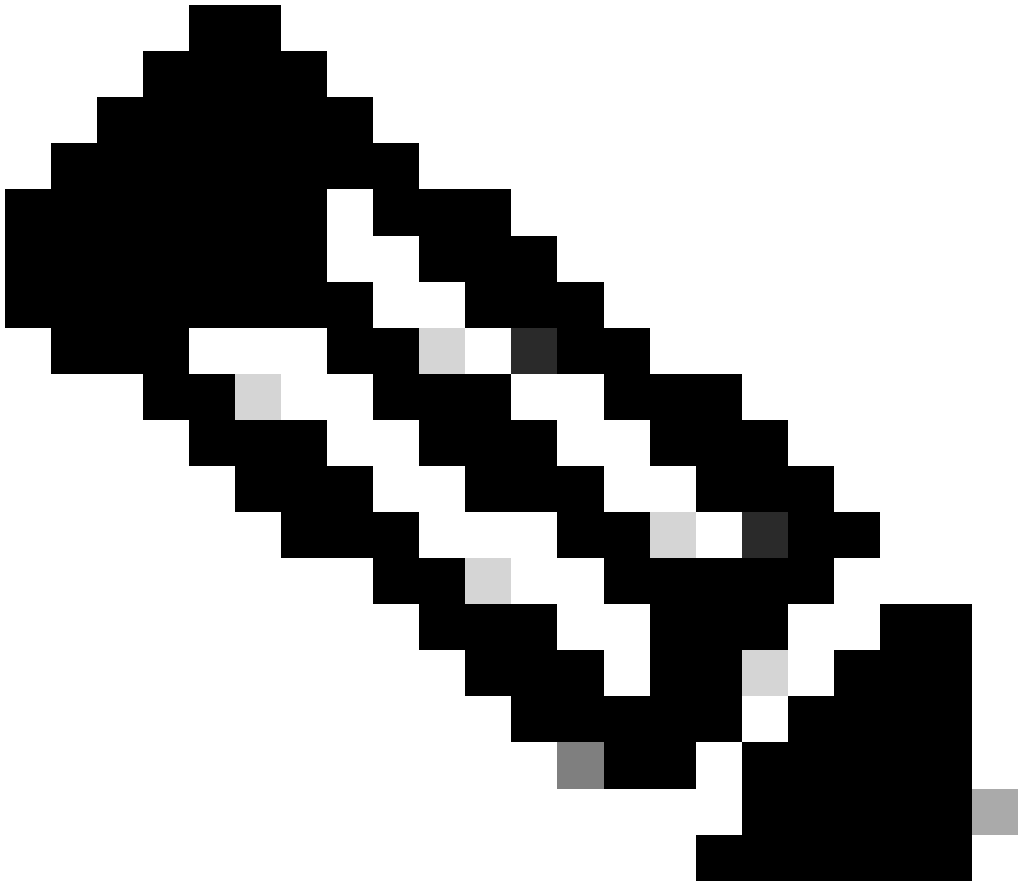
效能影響會根據啟用的封包追蹤選項而有所不同。資料包跟蹤僅影響被跟蹤的資料包（如與使用者配置條件匹配的資料包）的轉發效

能。配置要捕獲的資料包跟蹤的資訊越精細和詳細，對資源的影響就越大。

與任何故障排除一樣，最好採用迭代方法，僅在調試情況需要時才啟用更詳細的跟蹤選項。

QFP DRAM使用量可用以下公式估算：

需要的記憶體= （統計額外負荷） + pkts數量* （摘要大小+路徑資料大小+複製大小）



注意：如果**stats overhead**和**summary size**分別固定為2 KB和128 B，則**path data size**和**copy size**可由使用者配置。

相關資訊

- **Cisco ASR1000系列聚合系列路由器軟體配置指南-資料包跟蹤**
- **Cisco ASR1000系列服務路由器上的資料包丟棄**
- **思科技術支援與下載**

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。