

在ASR 9000中配置XR嵌入式Packet Tracer

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[使用的元件](#)

[背景資訊](#)

[XR嵌入式Packet Tracer限制和限制](#)

[Packet Tracer工作流程](#)

[設定](#)

[清除Packet Tracer計數器和條件](#)

[開始/停止資料包跟蹤](#)

[Packet Tracer條件](#)

[Packet Tracer條件 — 介面](#)

[Packet Tracer條件 — 偏移/值/掩碼](#)

[組態範例](#)

[相關資訊](#)

簡介

本檔案將介紹XR嵌入式Packet Tracer。它有助於跟蹤自定義資料包流，以進行服務驗證和故障排除。

必要條件

需求

XR Embedded Packet Tracer最初從Cisco IOS® XR版本7.1.2開始提供，ASR 9000系列支援它。計畫在未來更新中獲得其他XR產品系列的支援。

採用元件

XR Embedded Packet Tracer獨立於特定協定，並相容所有型別的單播和組播資料包。

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

XR Embedded Packet Tracer框架大大簡化了服務流驗證和資料包轉發問題的故障排除。

當在介面上啟用資料包跟蹤時，網路處理器(NP)會評估傳入資料包，以確定它們是否滿足定義的標準。如果資料包滿足指定的條件，則會將識別符號新增到其內部報頭。此識別符號有助於在路由器內跨資料路徑和點路徑所涉及的所有元件跟蹤資料包。

條件是指一組標準或規則，定義哪些資料包在通過路由器時可以跟蹤。這些情況有助於系統識別並監控特定資料包流，以便進行故障排除或服務驗證。

條件包括以下組成部分：

1. 實體介面：

- 指定資料包預期到達的網路介面。
- 範例： `packet-trace condition interface Gi0/0/0/1`

2. 偏移/值/掩碼三重：

- 定義匹配資料包報頭（或負載）的特定部分的標準。
- 這三個部分允許框架與協定無關，因為它們可以代表任何協定報頭的任何部分。

範例：

- Offset:位於資料包內（位元組偏移）。
- 值：該位置的預期值。
- 遮罩:要匹配的位精度。

XR嵌入式Packet Tracer限制和限制

XR 7.1.2版：

Lightspeed Plus、Lightspeed和Tomahawk線卡支援資料包標籤。

前面提到的線卡型別支援資料包跟蹤。

最多可以指定三個4個八位組的偏移/值/掩碼集。

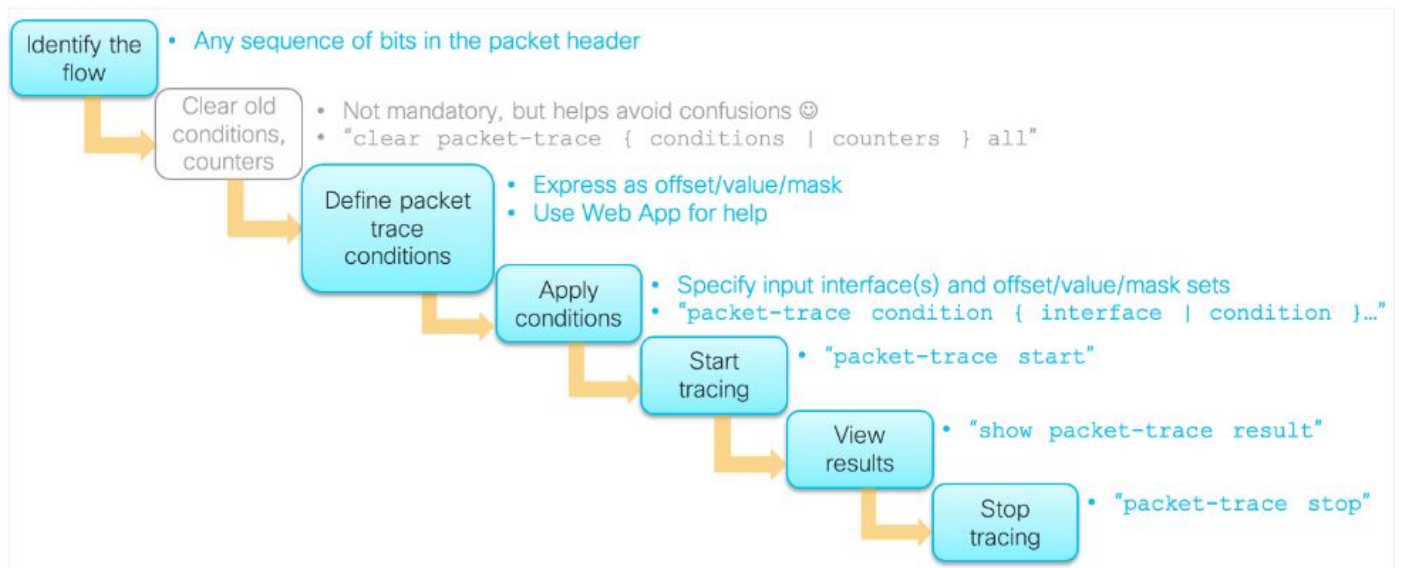
XR 7.5.2版：

Packet tracer會在設定條件時自動解析套件組成員

現在，您可以在SPP、NetIO、UDP、TCP中跟蹤分支路徑上的資料包

Packet Tracer工作流程

此圖說明Packet Tracer工作流的運行方式。



設定

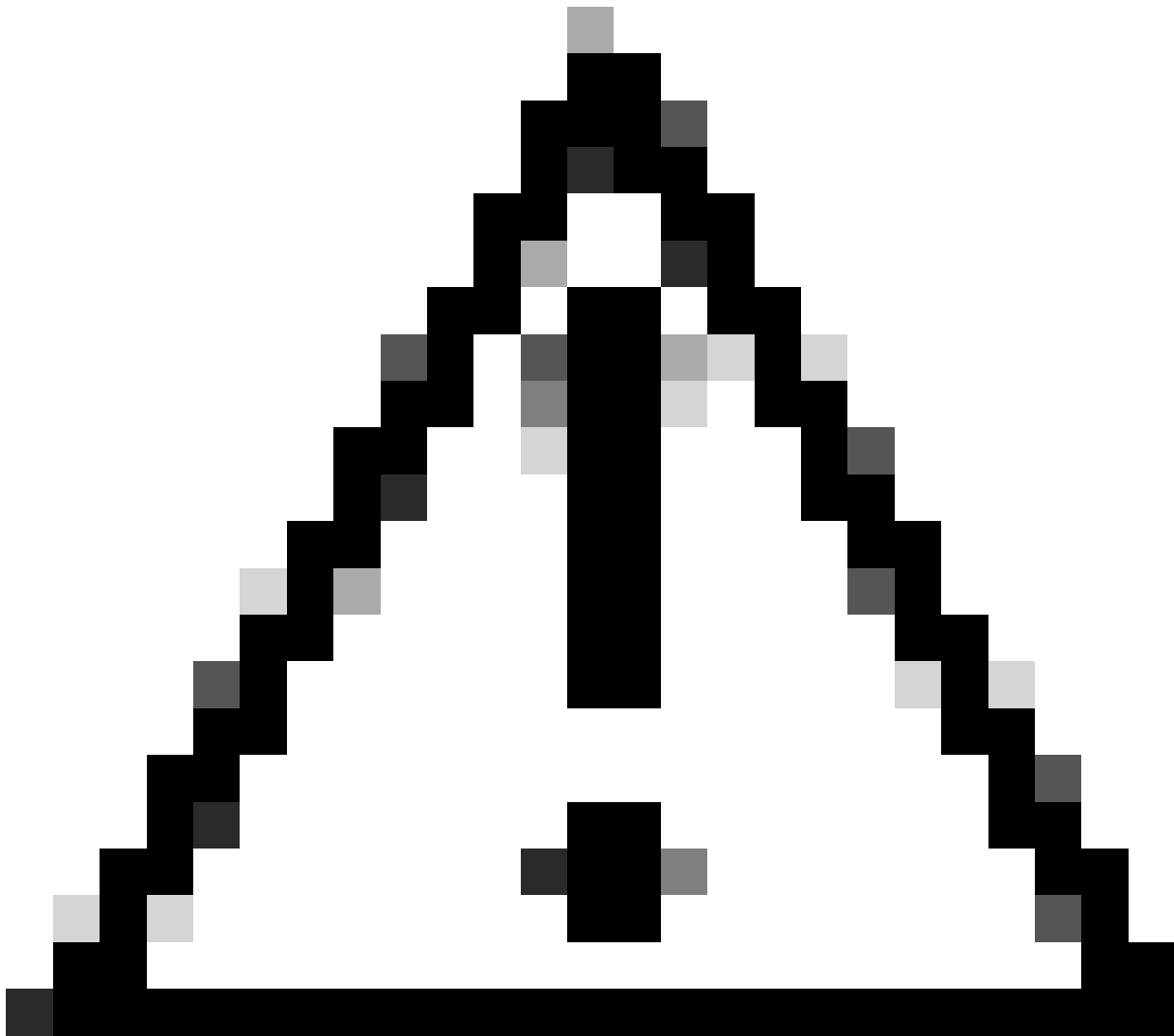
清除Packet Tracer計數器和條件

命令重置Packet Tracer計數器；Packet Tracer計數器可以隨時重置：

```
clear packet-trace counters all
```

命令要刪除所有Packet Tracer條件，請使用以下命令：

```
clear packet-trace conditions all
```



注意：根據設計，只有在資料包跟蹤處於非活動狀態時，才能清除Packet Tracer條件。

開始/停止資料包跟蹤

需要手動指定資料包跟蹤的開始和結束：

```
RP/0/RP0/CPU0:Device# packet-trace start
RP/0/RP0/CPU0:Device# packet-trace stop
```

Packet Tracer條件

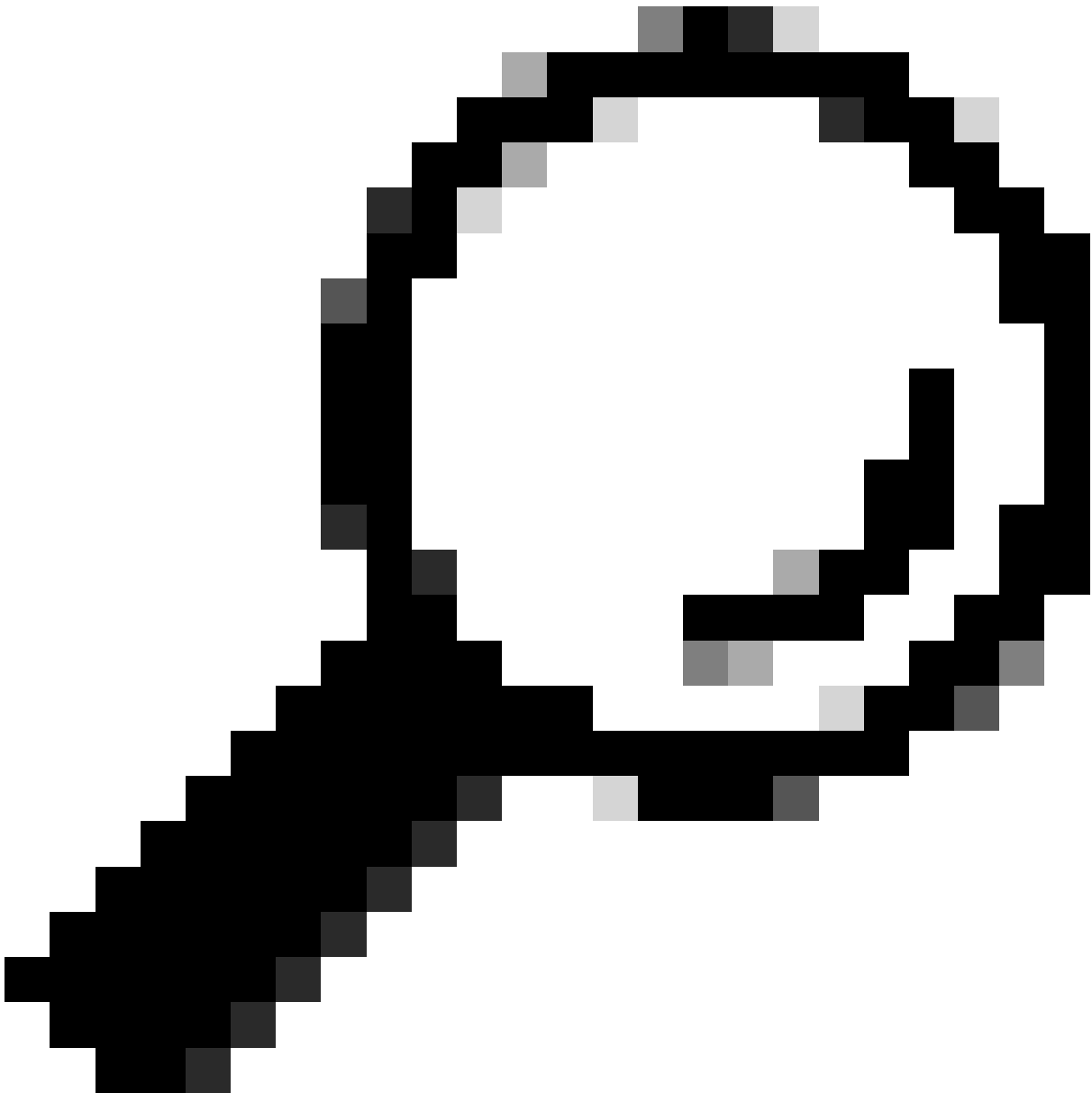
條件包括：

1. 物理介面：指示預期接收資料包的物理介面。

2. 偏移量/值/掩碼三重。幫助定義興趣流。

Packet Tracer條件 — 介面

```
RP/0/RP0/CPU0:Device#packet-trace condition interface GigE0/0/0/0
RP/0/RP0/CPU0:Device#packet-trace condition interface GigE0/0/0/1
```



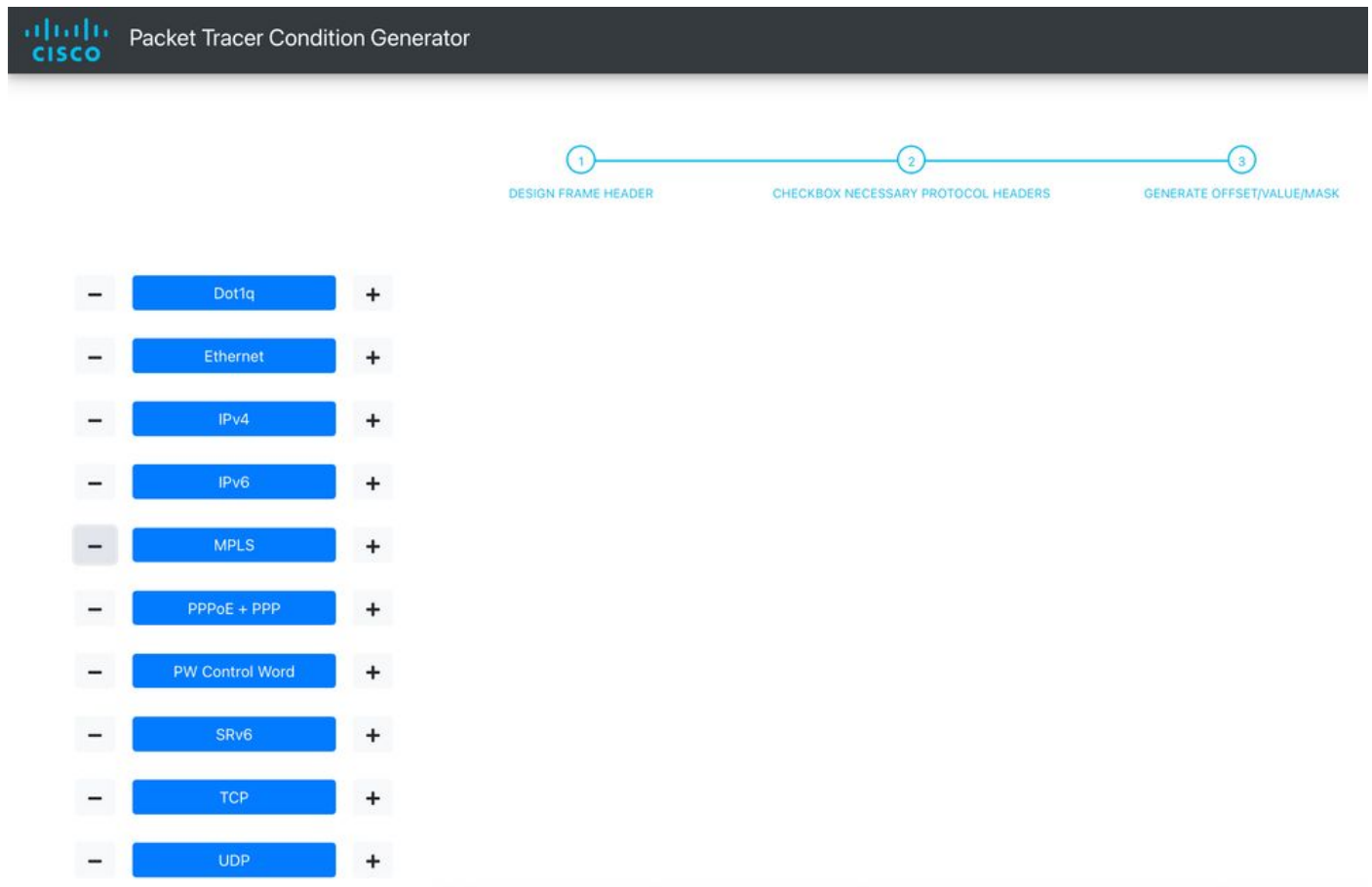
提示：在子介面上跟蹤時，「偏移/值/掩碼」規範必須考慮dot1q或QinQ封裝。

Packet Tracer條件 — 偏移/值/掩碼

「XR Packet Tracer Condition Generator Web App」提供了建立Packet Tracer條件的工具。

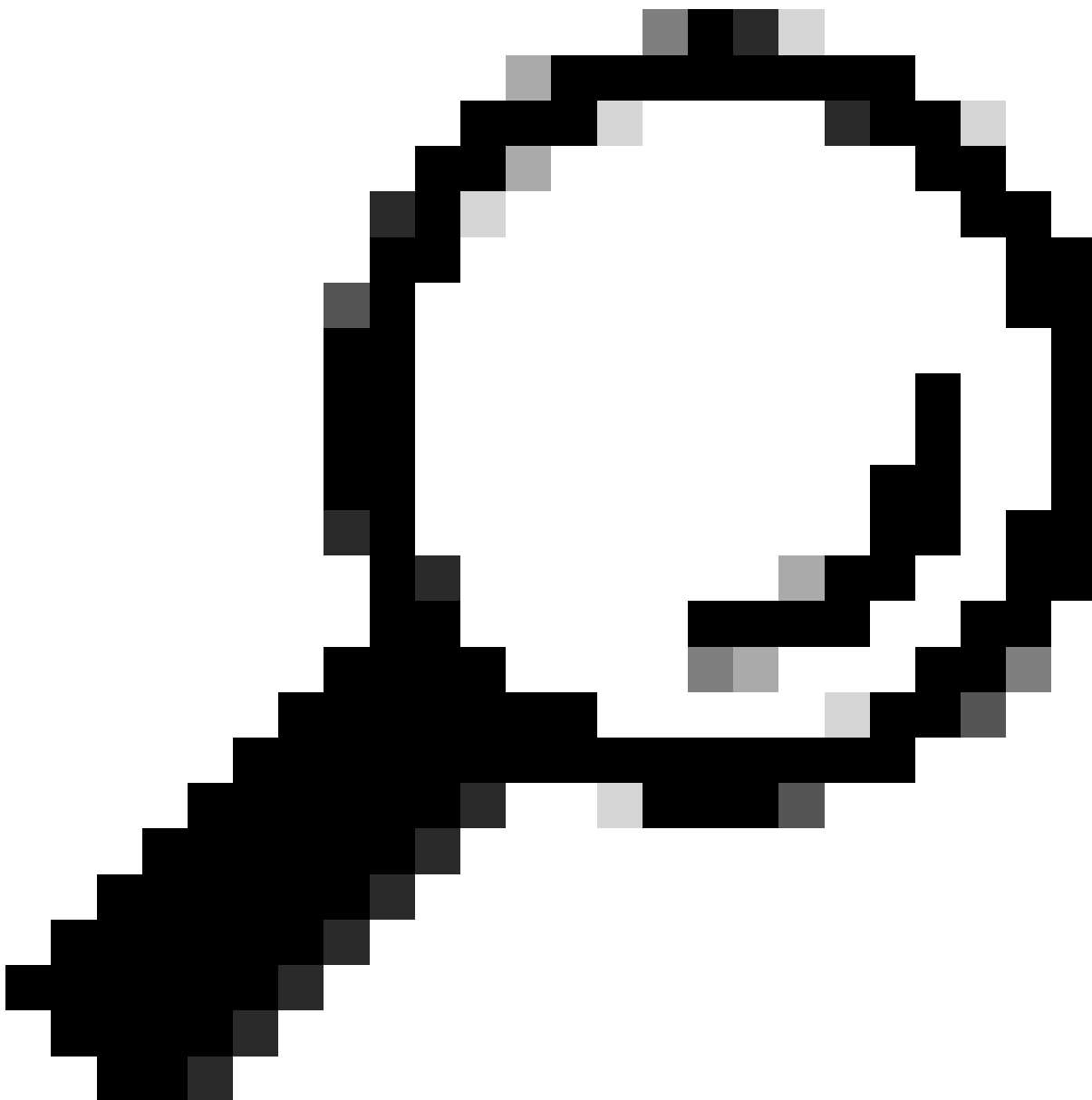
在GitHub上，可以以[XR Embedded Packet Tracer - Condition Generator](#)的名義訪問其原始碼和安裝指南。

使用此應用程式，您可以直觀地構建所需資料包流的協定棧，選擇相關層來定義條件，並輸入描述要跟蹤的特定流的值（帶有可選掩碼）。



Web App的登入頁顯示配置支援的協定報頭的清單。

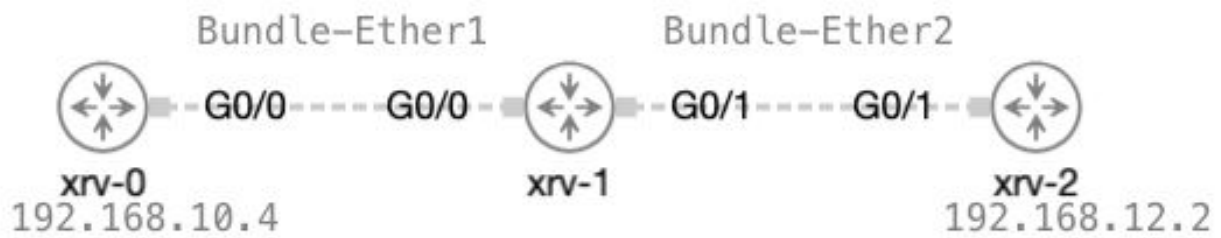
請確保在要匹配流量的報頭之前包含所有必要的報頭，因為偏移計算取決於報頭的順序。



提示：切記，如果正在使用PW控制字頭，請將其包含在內。

組態範例:

以下是拓撲示例。我們的目標是驗證資料包是否正確接收並通過XRV1裝置傳輸：



1. — 為要監控的特定介面設定packet-trace條件。

```
RP/0/RP0/CPU0:xrv-1#packet-trace condition interface Bundle-Ether1
RP/0/RP0/CPU0:xrv-1#packet-trace condition interface Bundle-Ether2
```

2. — 生成「偏移/值/掩碼」(Offset/Value/Mask)，選中要匹配的標題旁邊的覈取方塊。如有必要，可以選擇多個標頭。對於每個選定的報頭，相應的幀將顯示在右側。在幀中輸入所需的值和掩碼，然後按一下Submit按鈕完成配置。

3. — 將「偏移/值/遮罩」複製到剪貼簿後，使用它定義條件：

```
RP/0/RP0/CPU0:xrv-1#packet-trace condition 1 Offset 30 Value 0xc0a80a Mask 0xffffffff
RP/0/RP0/CPU0:xrv-1#packet-trace condition 5 Offset 34 Value 0xc0a80c Mask 0xffffffff
```

4. — 檢查資料包跟蹤狀態：

RP/0/RP0/CPU0:xrv-1#show packet-trace status

Packet Trace Master Process:

Buffered Conditions:

Interface Bundle-Ether1

Member GigE0/0/0/0

Interface Bundle-Ether2

Member GigE0/0/0/1

1 Offset 30 Value 0xc0a80a Mask 0xffffffff

5 Offset 34 Value 0xc0a80c Mask 0xffffffff

Status: Inactive

RP/0/RP0/CPU0:xrv-1#

RP/0/RP0/CPU0:xrv-1#show packet-trace status detail

Location: 0/0/CPU0

Available Counting Modules: 4

#1 SPP

Last errors:

#2 npu_server_lsp

Last errors:

#3 NETIO

Last errors:

#4 UDP

Last errors:

Available Marking Modules: 1

#1 npu_server_lsp

Interfaces: 0

Conditions: 0

Last errors:

Packet Trace Master Process:

Buffered Conditions:

Interface Bundle-Ether1

Member GigE0/0/0/0

Interface Bundle-Ether2

Member GigE0/0/0/1

1 Offset 30 Value 0xc0a80a Mask 0xffffffff

5 Offset 34 Value 0xc0a80c Mask 0xffffffff

Status: Inactive

Location: 0/RP0/CPU0

Available Counting Modules: 3

#1 SPP

Last errors:

#2 NETIO

Last errors:

#3 UDP

Last errors:

Available Marking Modules: 0

RP/0/RP0/CPU0:xrv-1#

5. — 啟動Packet Tracer:

```
RP/0/RP0/CPU0:xrv-1# packet-trace start
RP/0/RP0/CPU0:xrv-1#
RP/0/RP0/CPU0:xrv-1# show packet-trace status
```

```
-----
Packet Trace Master Process:
  Buffered Conditions:
    Interface Bundle-Ether1
      Member GigE0/0/0/0
    Interface Bundle-Ether2
      Member GigE0/0/0/1
      1 Offset 30 Value 0xc0a80a Mask 0xffffffff
      5 Offset 34 Value 0xc0a80c Mask 0xffffffff
  Status: Active
RP/0/RP0/CPU0:xrv-1#
```

6. — 讓我們等待幾分鐘，以便擷取流量：

7. — 檢查結果：

```
RP/0/RP0/CPU0:xrv-1#show packet-trace result
```

T: D - Drop counter; P - Pass counter

Location	Source	Counter	T	Last-Attribute
0/0/CPU0	NP0	PACKET_MARKED	P	GigE0_0_0_0
0/0/CPU0	NP0	PACKET_TO_FABRIC	P	
0/0/CPU0	NP0	PACKET_TO_PUNT	P	
0/0/CPU0	NP0	PACKET_FROM_FABRIC	P	
0/0/CPU0	NP0	PACKET_TO_INTERFACE	P	GigE0_0_0_1

```
RP/0/RP0/CPU0:xrv-1#
```

8. — 可以使用show packet-trace description 命令檢查在Packet tracer框架中註冊的所有計數器及其說明：

```
RP/0/RP0/CPU0:xrv-1#show packet-trace descriptions
```

NP0	PACKET_MARKED	M	Marked from ingress interface
NP0	PACKET_FROM_INJECT	P	Injected from linecard CPU
NP0	PACKET_FROM_FAB_INJECT	P	Injected from fabric
NP0	PACKET_ING_DROP	D	Dropped on ingress
NP0	PACKET_TO_FABRIC	P	Sent to router fabric
NP0	PACKET_TO_PUNT	P	Punted to linecard for CPU handling
NP0	PACKET_FROM_FABRIC	P	From router fabric
NP0	PACKET_EGR_DROP	D	Dropped on egress
NP0	PACKET_TO_INTERFACE	P	Packet sent to network interface

```
RP/0/RP0/CPU0:xrv-1#
```

9. — 停止資料包跟蹤：

```
RP/0/RP0/CPU0:xrv-1#packet-trace stop
```

相關資訊

[XR嵌入式Packet Tracer](#)

[思科技術支援與下載](#)

[瞭解ASR 9000系列線路卡型別](#)

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）。