在Catalyst 9000系列交換器上設定和驗證Netflow、AVC和ETA

目錄

簡介

本文說明如何配置和驗證NetFlow、應用可視性與可控性(AVC)以及加密流量分析(ETA)。

必要條件

需求

思科建議您瞭解以下主題：

- Netflow
- AVC
- 埃塔

採用元件

本檔案中的資訊是根據執行Cisco IOS XE軟體16.12.4的Catalyst 9300交換器。

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。

相關產品

本文件也適用於以下硬體和軟體版本：

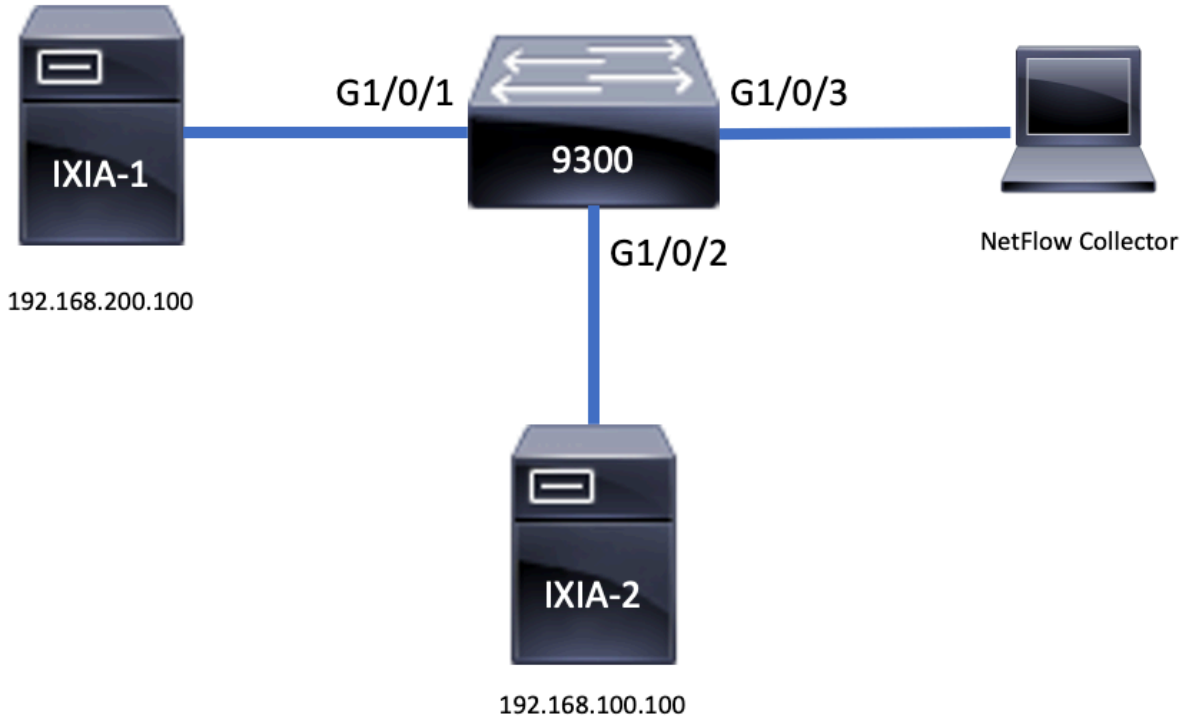
- 9200
- 9400
- 9500
- 9600
- Cisco IOS XE 16.12及更高版本

背景資訊

- Flexible NetFlow是下一代流量技術，它收集和測量資料，使網路中的所有路由器或交換機成為遙測源。
- Flexible NetFlow允許進行極其精細和準確的流量測量以及高級匯聚流量收集。
- Flexible NetFlow使用流為記帳、網路監控和網路規劃提供統計資料。
- 流是到達來源介面的單向封包流，其關鍵字的值相同。鍵是資料包中欄位的標識值。您可以通過流記錄建立流，以定義流的唯一鍵。

> **附註**：平台(fed)命令可能有所不同。命令可以是「**show platform fed <active|standby>**」和「**show platform fed switch <active|standby>**」。 如果示例中註明的語法未解析出，請嘗試變體。

網路圖表



設定

元件

NetFlow配置由三個可**一起使用的**主要元件組成，它們有多種變體用於執行流量分析和資料匯出。

流記錄

- 記錄是鍵欄位和非鍵欄位的組合。將Flexible NetFlow記錄分配給Flexible NetFlow流監控器，以定義用於儲存流資料的快取。
- Flexible NetFlow包括可用於監控流量的多個預定義記錄。
- Flexible NetFlow還允許通過指定鍵和非鍵欄位來為Flexible NetFlow流量監控器快取定義自定義記錄，以便根據您的特定要求自定義資料收集。

如示例所示，流記錄配置詳細資訊：

```
flow record TAC-RECORD-IN
match flow direction
match ipv4 source address
match interface input
match ipv4 destination address
match ipv4 protocol
collect counter packets long
collect counter bytes long
collect timestamp absolute last
collect transport tcp flags

flow record TAC-RECORD-OUT
match flow direction
match interface output
```

```
match ipv4 source address
match ipv4 destination address
match ipv4 protocol
collect counter packets long
collect counter bytes long
collect timestamp absolute last
collect transport tcp flags
```

流匯出器

- 流匯出器用於將流監控器快取中的資料匯出到遠端系統（充當NetFlow收集器的伺服器）進行分析和儲存。
- 流匯出器分配給流監控器，以便為流監控器提供資料匯出功能。

如示例所示，流匯出器配置詳細資訊：

```
flow exporter TAC-EXPORT
destination 192.168.69.2
source Vlan69
```

流量監控器

- 流量監控器是應用於介面以執行網路流量監控的Flexible NetFlow元件。
- 流資料從網路流量中收集，並在進程運行時新增到流監控器快取中。該流程基於流記錄中的關鍵欄位和非關鍵欄位。

如示例所示，流量監控器配置詳細資訊：

```
flow monitor TAC-MONITOR-IN
exporter TAC-EXPORT
record TAC-RECORD-IN

flow monitor TAC-MONITOR-OUT
exporter TAC-EXPORT
record TAC-RECORD-OUT

Switch#show run int g1/0/1
Building configuration...

Current configuration : 185 bytes
!
interface GigabitEthernet1/0/1
switchport access vlan 42
switchport mode access
ip flow monitor TAC-MONITOR-IN input
ip flow monitor TAC-MONITOR-OUT output
load-interval 30
end
```

流量取樣器（可選）

- 流量取樣器作為路由器配置中的獨立元件建立。
- 流量取樣器限制選擇進行分析的資料包數量，以減少使用Flexible NetFlow的裝置上的負載。
- 流量取樣器用於減少使用通過限制選擇用於分析的資料包數量實現的Flexible NetFlow的裝置上的負載。
- 流量取樣器可以交換路由器效能的準確性。如果流監控器分析的資料包數量減少，則儲存在流

監控器快取中的資訊的準確性會受到影響。

如示例所示，示例流量取樣器配置：

```
sampler SAMPLE-TAC
description Sample at 50%
mode random 1 out-of 2

Switch(config)#interface GigabitEthernet1/0/1
Switch(config-if)#ip flow monitor TAC-MONITOR-IN sampler SAMPLE-TAC input
Switch(config-if)#end
```

限制

- 完整Flexible NetFlow需要DNA附加許可證，否則取樣NetFlow僅可用。
- 流量匯出器無法使用管理埠作為源。

這不是一個包含清單，請參考配置指南，瞭解相應的平台和代碼。

驗證

獨立於平台的驗證

驗證配置並確認所需的NetFlow元件存在：

1. **流記錄**
2. **流匯出器**
3. **流量監控器**
4. **流量取樣器（可選）**

提示：要檢視一個命令中的流記錄、流匯出器和流監控器輸出，請運行「show running-config flow monitor <flow monitor name> expand」

如示例所示，繫結到輸入方向的流量監控器及其關聯的元件：

```
Switch#show running-config flow monitor TAC-MONITOR-IN expand
Current configuration:
!
flow record TAC-RECORD-IN
 match ipv4 protocol
 match ipv4 source address
 match ipv4 destination address
 match interface input
 match flow direction
 collect transport tcp flags
 collect counter bytes long
 collect counter packets long
 collect timestamp absolute last
!
flow exporter TAC-EXPORT
 destination 192.168.69.2
 source Vlan69
!
flow monitor TAC-MONITOR-IN
 exporter TAC-EXPORT
```

```
 record TAC-RECORD-IN
!
```

如示例所示，與輸出方向及其關聯元件繫結的流量監控器：

```
Switch#show run flow monitor TAC-MONITOR-OUT expand
Current configuration:
!
flow record TAC-RECORD-OUT
 match ipv4 protocol
 match ipv4 source address
 match ipv4 destination address
 match interface output
 match flow direction
 collect transport tcp flags
 collect counter bytes long
 collect counter packets long
 collect timestamp absolute last
!
flow exporter TAC-EXPORT
 destination 192.168.69.2
 source Vlan69
!
flow monitor TAC-MONITOR-OUT
 exporter TAC-EXPORT
 record TAC-RECORD-OUT
!
```

運行命令「**show flow monitor <flow monitor name>」 statistics。**此輸出有助於確認已記錄資料：

```
Switch#show flow monitor TAC-MONITOR-IN statistics
  Cache type:                               Normal (Platform cache)
  Cache size:                                10000
  Current entries:                               1

  Flows added:                                   1
  Flows aged:                                    0
```

運行**show flow monitor <flow monitor name>快取命令**，確認NetFlow快取有輸出：

```
Switch#show flow monitor TAC-MONITOR-IN cache
  Cache type:                               Normal (Platform cache)
  Cache size:                                10000
  Current entries:                               1

  Flows added:                                   1
  Flows aged:                                    0

IPV4 SOURCE ADDRESS:       192.168.200.100
IPV4 DESTINATION ADDRESS:  192.168.100.100
INTERFACE INPUT:           Gi1/0/1
FLOW DIRECTION:            Input
IP PROTOCOL:               17
tcp flags:                 0x00
counter bytes long:        4606617470
counter packets long:      25311085
timestamp abs last:        22:44:48.579
```

運行命令「**show flow exporter <exporter name> statistics」，**確認匯出器傳送了資料包：

```
Switch#show flow exporter TAC-EXPORT statistics
Flow Exporter TAC-EXPORT:
  Packet send statistics (last cleared 00:08:38 ago):
    Successfully sent:         2                     (24 bytes)

  Client send statistics:
    Client: Flow Monitor TAC-MONITOR-IN
      Records added:           0
      Bytes added:             12
        - sent:                12

    Client: Flow Monitor TAC-MONITOR-OUT
      Records added:           0
      Bytes added:             12
        - sent:                12
```

平台相關驗證

NetFlow初始化 — NFL分割槽表

- NetFlow分割槽針對不同的功能進行初始化，每個方向有16個分割槽（輸入與輸出）。
- NetFlow分割槽表配置分為全域性銀行分配，進一步細分為入口和出口流銀行。

關鍵欄位

- 分割槽數
- 分割槽啟用狀態
- 分割槽限制
- 當前分割槽使用情況

要檢視NetFlow分割槽表，可以運行命令「show platform software fed switch active|standby|member| fnf sw-table-sic <asic number> shadow 0」

> **附註**：建立的流特定於交換機和asic核心。需要相應地指定交換器編號（主用、備用等）。輸入的ASIC編號繫結到各個介面，使用「show platform software fed switch active|standby|member ifm mappings」確定與該介面對應的ASIC。對於陰影選項，始終使用「0」。

```
Switch#show platform software fed switch active fnf sw-table-sizes asic 0 shadow 0

--------------------------------
     Global Bank Allocation
--------------------------------
Ingress Banks : Bank 0 Bank 1
Egress Banks  : Bank 2 Bank 3
--------------------------------
    Global flow table Info                                  <--- Provides the number of entries
used per direction
INGRESS    usedBankEntry           0  usedOvfTcamEntry     0
 EGRESS    usedBankEntry           0  usedOvfTcamEntry     0
--------------------------------
    Flows Statistics
INGRESS    TotalSeen=0 MaxEntries=0 MaxOverflow=0
EGRESS     TotalSeen=0 MaxEntries=0 MaxOverflow=0

--------------------------------
```

```
       Partition Table
--------------------------------
## Dir  Limit  CurrFlowCount  OverFlowCount  MonitoringEnabled
 0 ING      0              0              0                  0
 1 ING  16640              0              0                  1          <-- Current flow count in hardware
 2 ING      0              0              0                  0
 3 ING  16640              0              0                  0
 4 ING      0              0              0                  0
 5 ING   8192              0              0                  1
 6 ING      0              0              0                  0
 7 ING      0              0              0                  0
 8 ING      0              0              0                  0
 9 ING      0              0              0                  0
10 ING      0              0              0                  0
11 ING      0              0              0                  0
12 ING      0              0              0                  0
13 ING      0              0              0                  0
14 ING      0              0              0                  0
15 ING      0              0              0                  0
 0 EGR      0              0              0                  0
 1 EGR  16640              0              0                  1          <-- Current flow count in hardware
 2 EGR      0              0              0                  0
 3 EGR  16640              0              0                  0
 4 EGR      0              0              0                  0
 5 EGR   8192              0              0                  1
 6 EGR      0              0              0                  0
 7 EGR      0              0              0                  0
 8 EGR      0              0              0                  0
 9 EGR      0              0              0                  0
10 EGR      0              0              0                  0
11 EGR      0              0              0                  0
12 EGR      0              0              0                  0
13 EGR      0              0              0                  0
14 EGR      0              0              0                  0
15 EGR      0              0              0                  0
```

流量監控器

流量監控器配置包括：

1. NetFlow ACL配置，這會導致在ACL TCAM表中建立條目。

ACL TCAM條目由以下部分組成：

- 查詢匹配鍵
- 用於NetFlow查詢的結果引數，其中包括：
  配置檔案IDNetFlow ID

2.流掩碼配置，這會導致NflLookupTable和NflFlowMaskTable中建立一個條目。

- 通過NetFlow ACL結果引數索引，以查詢netflow查詢的流掩碼

NetFlow ACL

要檢視NetFlow ACL配置，請運行命令「**show platform hardware fed switch active fwd-asic resource tcam table nfl_acl asic <asic number>**

提示：如果有埠ACL(PACL)，則在該介面對映到的ASIC上建立該條目。在路由器ACL(RACL)的情況下，條目存在於所有ASIC上。

- 在此輸出中，有NFCMD0和NFCMD1是4位值。為了計算配置檔案ID，請將值轉換為二進位制。
- 在此輸出中，NFCMD0為1,NFCMD1為2。轉換為二進位制時：000100010
- 在Cisco IOS-XE 16.12中，在組合的8位內，前4位是配置檔案ID，第7位表示已啟用查詢。因此在示例**0001**0010中，配置檔案ID為1。
- 在Cisco IOS XE 16.11及更舊版本的代碼中，在組合的8位內，前6位是配置檔案ID，第7位表示已啟用查詢。在本例中，**000100**10，配置檔案ID為4。

```
Switch#show platform hardware fed switch active fwd-asic resource tcam table nfl_acl asic 0
Printing entries for region INGRESS_NFL_ACL_CONTROL (308) type 6 asic 0
=======================================================
Printing entries for region INGRESS_NFL_ACL_GACL (309) type 6 asic 0
=======================================================
Printing entries for region INGRESS_NFL_ACL_PACL (310) type 6 asic 0
=======================================================
TAQ-2 Index-32 (A:0,C:0) Valid StartF-1 StartA-1 SkipF-0 SkipA-0
Input IPv4 NFL PACL

Labels  Port   Vlan   L3If   Group
M:      00ff   0000   0000   0000
V:      0001   0000   0000   0000

   vcuResults l3Len l3Pro l3Tos SrcAddr  DstAddr  mtrid vrfid  SH
M:  00000000   0000  00    00  00000000 00000000  00   0000   0000
V:  00000000   0000  00    00  00000000 00000000  00   0000   0000

   RMAC RA MEn IPOPT MF NFF DF SO DPT TM DSEn  l3m
M:  0    0  0    0    0  0   0  0  0   0  0  0
V:  0    0  0    0    0  0   0  0  0   0  0  0

   SrcPort DstPortIITypeCode TCPFlags TTL ISBM QosLabel ReQOS S_P2P D_P2P
M:  0000    0000              00    00    0000    00      0     0     0
V:  0000    0000              00    00    0000    00      0     0     0

   SgEn SgLabel    AuthBehaviorTag  l2srcMiss  l2dstMiss ipTtl SgaclDeny
M:  0    000000    0  0   0 0     0
V:  0    000000    0  0   0 0     0

NFCMD0 NFCMD1 SMPLR LKP1 LKP2 PID QOSPRI MQLBL MPLPRO LUT0PRI CPUCOPY
     1      2     0    1    0   0      0     0      0 0x0000f       0
Start/Skip Word: 0x00000003
Start Feature, Terminate

---------------------------------------
Printing entries for region INGRESS_NFL_ACL_VACL (311) type 6 asic 0
=======================================================
Printing entries for region INGRESS_NFL_ACL_RACL (312) type 6 asic 0
=======================================================
Printing entries for region INGRESS_NFL_ACL_SSID (313) type 6 asic 0
=======================================================
Printing entries for region INGRESS_NFL_CATCHALL (314) type 6 asic 0
=======================================================
TAQ-2 Index-224 (A:0,C:0) Valid StartF-1 StartA-1 SkipF-0 SkipA-0
Input IPv4 NFL RACL

Labels  Port   Vlan   L3If   Group
```

```
M:      0000   0000   0000   0000
V:      0000   0000   0000   0000

   vcuResults l3Len l3Pro l3Tos SrcAddr  DstAddr  mtrid vrfid  SH
M:  00000000   0000  00    00  00000000 00000000  00   0000   0000
V:  00000000   0000  00    00  00000000 00000000  00   0000   0000

   RMAC RA MEn IPOPT MF NFF DF SO DPT TM DSEn  l3m
M:  0    0  0    0    0  0   0  0  0   0  0  0
V:  0    0  0    0    0  0   0  0  0   0  0  0

   SrcPort DstPortIITypeCode TCPFlags TTL ISBM QosLabel ReQOS S_P2P D_P2P
M:  0000    0000              00    00    0000    00      0     0     0
V:  0000    0000              00    00    0000    00      0     0     0

   SgEn SgLabel    AuthBehaviorTag  l2srcMiss  l2dstMiss ipTtl SgaclDeny
M:  0    000000    0  0   0 0     0
V:  0    000000    0  0   0 0     0

NFCMD0 NFCMD1 SMPLR LKP1 LKP2 PID QOSPRI MQLBL MPLPRO LUT0PRI CPUCOPY
     0      0     0    0    0   0      0     0      0 0x00000       0
Start/Skip Word: 0x00000003
Start Feature, Terminate

----------------------------------------
TAQ-2 Index-225 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Input IPv4 NFL PACL

Labels  Port   Vlan   L3If   Group
M:      0000   0000   0000   0000
V:      0000   0000   0000   0000

   vcuResults l3Len l3Pro l3Tos SrcAddr  DstAddr  mtrid vrfid  SH
M:  00000000   0000  00    00  00000000 00000000  00   0000   0000
V:  00000000   0000  00    00  00000000 00000000  00   0000   0000

   RMAC RA MEn IPOPT MF NFF DF SO DPT TM DSEn  l3m
M:  0    0  0    0    0  0   0  0  0   0  0  0
V:  0    0  0    0    0  0   0  0  0   0  0  0

   SrcPort DstPortIITypeCode TCPFlags TTL ISBM QosLabel ReQOS S_P2P D_P2P
M:  0000    0000              00    00    0000    00      0     0     0
V:  0000    0000              00    00    0000    00      0     0     0

   SgEn SgLabel    AuthBehaviorTag  l2srcMiss  l2dstMiss ipTtl SgaclDeny
M:  0    000000    0  0   0 0     0
V:  0    000000    0  0   0 0     0

NFCMD0 NFCMD1 SMPLR LKP1 LKP2 PID QOSPRI MQLBL MPLPRO LUT0PRI CPUCOPY
     0      0     0    0    0   0      0     0      0 0x00000       0
Start/Skip Word: 0x00000000
No Start, Terminate

----------------------------------------
TAQ-2 Index-226 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Input IPv6 NFL PACL

Labels  Port   Vlan   L3If   Group
Mask    0x0000 0x0000 0x0000 0x0000
Value   0x0000 0x0000 0x0000 0x0000

vcuResult dstAddr0 dstAddr1 dstAddr2 dstAddr3 srcAddr0
00000000  00000000 00000000 00000000 00000000 00000000
00000000  00000000 00000000 00000000 00000000 00000000
```

```
srcAddr1 srcAddr2 srcAddr3 TC HL l3Len fLabel vrfId toUs
00000000 00000000 00000000 00 00  0000  00000  000   0
00000000 00000000 00000000 00 00  0000  00000  000   0

l3Pro mtrId AE FE RE HE MF NFF SO IPOPT RA MEn RMAC DPT TMP l3m
 00    00   0  0  0  0  0   0  0    0   0   0   0    0   0    0
 00    00   0  0  0  0  0   0  0    0   0   0   0    0   0    0

DSE srcPort dstPortIITypeCode tcpFlags IIPresent cZId dstZId
 0   0000    0000      00      00     00     00
 0   0000    0000      00      00     00     00

v6RT AH ESP mREn ReQOS QosLabel PRole VRole  AuthBehaviorTag
M: 0   0   0    0    0      00         0     0 0
V: 0   0   0    0    0      00         0     0 0

   SgEn SgLabel
M:  0    000000
V:  0    000000

NFCMD0 NFCMD1 SMPLR LKP1 LKP2 PID QOSPRI MQLBL MPLPRO LUT0PRI CPUCOPY
     0      0     0    0    0   0      0     0      0 0x00000       0
Start/Skip Word: 0x00000000
No Start, Terminate

----------------------------------------
TAQ-2 Index-228 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
conversion to string vmr l2p not supported
----------------------------------------
TAQ-2 Index-230 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Input MAC NFL PACL

Labels  Port   Vlan   L3If   Group
M:      0000   0000   0000   0000
V:      0000   0000   0000   0000

   arpSrcHwAddr  arpDestHwAddr  arpSrcIpAddr  arpTargetIp  arpOperation
M: 000000000000  000000000000   00000000       00000000     0000
V: 000000000000  000000000000   00000000       00000000     0000


   TRUST  SNOOP  SVALID  DVALID
M:    0     0      0      0
V:    0     0      0      0


   arpHardwareLength  arpHardwareType  arpProtocolLength  arpProtocolType
M:   00000000          00000000            00000000          00000000
V:   00000000          00000000            00000000          00000000


   VlanId l2Encap l2Protocol cosCFI   srcMAC      dstMAC    ISBM  QosLabel
M:  000    0        0000     0     000000000000 000000000000 00     00
V:  000    0        0000     0     000000000000 000000000000 00     00


  ReQOS isSnap  isLLC  AuthBehaviorTag
M:  0     0      0       0
V:  0     0      0        0

NFCMD0 NFCMD1 SMPLR LKP1 LKP2 PID QOSPRI MQLBL MPLPRO LUT0PRI CPUCOPY
     0      0     0    0    0   0      0     0      0 0x00000       0
Start/Skip Word: 0x00000000
```

```
No Start, Terminate

---------------------------------------
```

流量掩碼

運行命令「show platform software fed switch active|standby|member fnf fmask-entry asic <asic number> entry 1」以檢視硬體中安裝了流掩碼。關鍵欄位清單的數量也可以在此處找到。

```
Switch#show platform software fed switch active fnf fmask-entry asic 1 entry 1

---------------------------------
mask0_valid : 1
Mask hdl0   : 1
Profile ID  : 0
Feature 0   : 148
Fmsk0 RefCnt: 1
Mask M1     :
[511:256] => :00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
[255:000] => :FFFFFFFF 00000000 FFFFFFFF 03FF0000 00000000 00FF0000 00000000 C00000FF

Mask M2     :

Key Map     :

Source   Field-Id   Size   NumPFields   Pfields
 002       090       04        01       (0 1 1 1)
 002       091       04        01       (0 1 1 0)
 002       000       01        01       (0 1 0 7)
 000       056       08        01       (0 0 2 4)
 001       011       11        04       (0 0 0 1) (0 0 0 0) (0 1 0 6) (0 0 2 0)
 000       067       32        01       (0 1 12 0)
 000       068       32        01       (0 1 12 2)
```

流統計資料和時間戳解除安裝資料

運行命令「show platform software fed switch active fnf flow-record asic <asic number> start-index <index number> num-flows <number of flows>」以檢視netflow統計資訊以及時間戳

```
Switch#show platform software fed switch active fnf flow-record asic 1 start-index 1 num-flows 1
1 flows starting at 1 for asic 1:-------------------------------------------------
Idx 996 :
{90, ALR_INGRESS_NET_FLOW_ACL_LOOKUP_TYPE1 = 0x01}
{91, ALR_INGRESS_NET_FLOW_ACL_LOOKUP_TYPE2 = 0x01}
{0, ALR_INGRESS_NFL_SPECIAL1 = 0x00}
{56, PHF_INGRESS_L3_PROTOCOL = 0x11}
{11 PAD-UNK = 0x0000}
{67, PHF_INGRESS_IPV4_DEST_ADDRESS = 0xc0a86464}
{68, PHF_INGRESS_IPV4_SRC_ADDRESS = 0xc0a8c864}
FirstSeen = 0x4b2f, LastSeen = 0x4c59, sysUptime = 0x4c9d
PKT Count = 0x000000000102d5df, L2ByteCount = 0x00000000ca371638


Switch#show platform software fed switch active fnf flow-record asic 1 start-index 1 num-flows 1
1 flows starting at 1 for asic 1:-------------------------------------------------
Idx 996 :
{90, ALR_INGRESS_NET_FLOW_ACL_LOOKUP_TYPE1 = 0x01}
{91, ALR_INGRESS_NET_FLOW_ACL_LOOKUP_TYPE2 = 0x01}
{0, ALR_INGRESS_NFL_SPECIAL1 = 0x00}
{56, PHF_INGRESS_L3_PROTOCOL = 0x11}
```

```
{11 PAD-UNK = 0x0000}
{67, PHF_INGRESS_IPV4_DEST_ADDRESS = 0xc0a86464}
{68, PHF_INGRESS_IPV4_SRC_ADDRESS = 0xc0a8c864}
FirstSeen = 0x4b2f, LastSeen = 0x4c5b, sysUptime = 0x4c9f
PKT Count = 0x0000000001050682, L2ByteCount = 0x00000000cbed1590
```

應用可視性與可控性(AVC)

背景資訊

- 應用可視性與可控性(AVC)是一種解決方案，它利用基於網路的識別版本2(**NBAR2**)、**NetFlow V9**以及各種報告和管理工具(**Cisco Prime**)來幫助通過深度資料包檢測(DPI)對應用進行分類。
- 可在獨立交換器或交換器堆疊的有線存取連線埠上設定AVC。
- AVC也可用於思科無線控制器，以根據DPI識別應用，然後將其標籤為特定DSCP值。它還可以收集各種無線效能指標，如應用和客戶端方面的頻寬使用情況。

效能和規模

效能：每個交換機成員能夠以低於50%的CPU利用率每秒處理500個連線(CPS)。在此速率之外，AVC服務沒有保證。

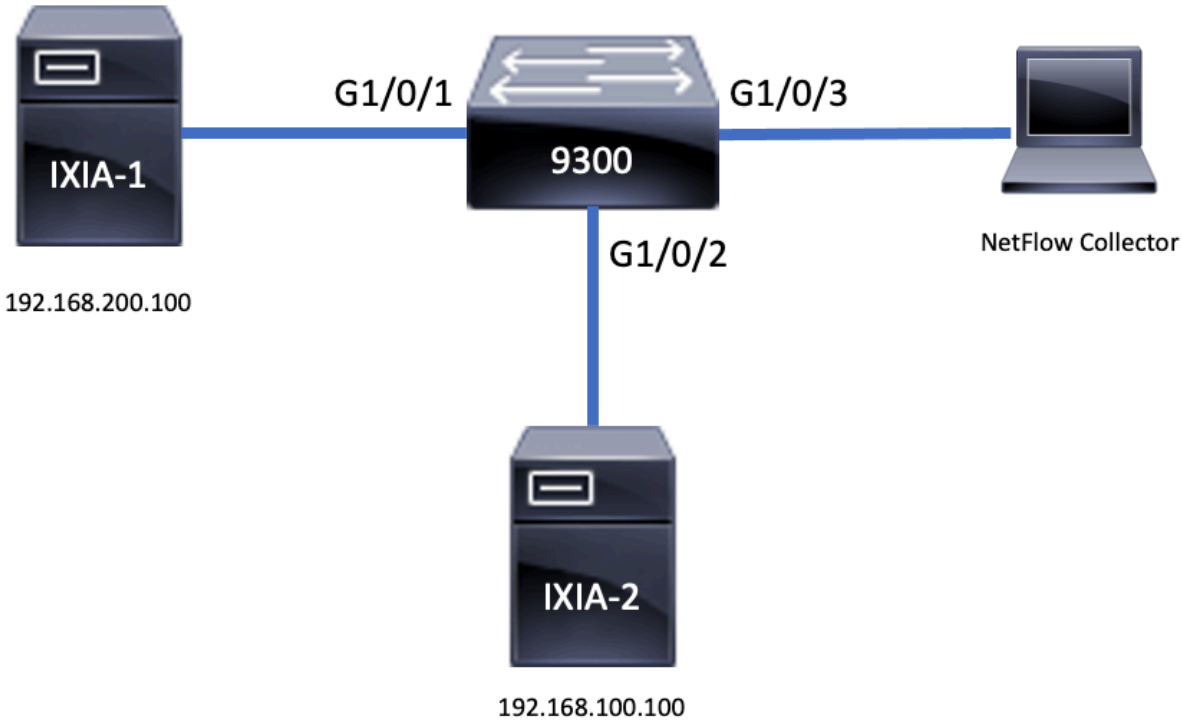
擴展：能夠處理每24個接入埠多達5000個雙向流（每個接入埠大約200個流）。

有線AVC限制

- 不能在同一介面上同時配置AVC和加密流量分析(ETA)。
- 只有單點傳播IPv4(TCP/UDP)流量支援封包分類。
- 僅在有線物理埠上支援基於NBAR的QoS策略配置。這包括第2層接入埠和中繼埠以及第3層路由埠。
- 埠通道成員、交換機虛擬介面(SVI)或子介面不支援基於NBAR的QoS策略配置。
- 基於NBAR2的分**類器(匹配協定)**，僅支援標籤和策略的QoS操作。
- 在所有策略中，「Match protocol」限製為255個不同的協定（8位硬體限制）

附註：這不是所有限制的完整清單，請參考適用於您的平台和代碼版本的相應AVC配置指南。

網路圖表



元件

AVC配置由構成解決方案的三個主要元件組成：

可視性：通訊協定探索

- 協定發現通過NBAR實現，NBAR提供每個介面、方向和應用程式位元組/資料包的統計資訊。
- 通過介面配置為特定介面啟用協定發現：**ip nbar protocol-discovery**

如輸出所示，如何啟用協定發現：

```
Switch(config)#interface fi4/0/5
Switch(config-if)#ip nbar protocol-discovery
Switch(config-if)#exit

Switch#show run int fi4/0/5
Building configuration...

Current configuration : 70 bytes
!
interface FiveGigabitEthernet4/0/5
ip nbar protocol-discovery
end
```

控制：基於應用的QoS

與傳統QoS（在IP地址和UDP/TCP埠上匹配）相比，AVC通過基於應用的QoS實現更精細的控制，允許您在應用上匹配，並通過標籤和策略等QoS操作提供更精細的控制。

- 對聚合流量（而不是每個流）執行操作
- 基於應用的QoS通過建立類對映、匹配協定以及建立策略對映來實現。
- 基於應用的QoS策略附加到介面。

如輸出所示，基於應用的QoS的配置示例：

```
Switch(config)#class-map WEBEX
Switch(config-cmap)#match protocol webex-media
Switch(config)#end

Switch(config)#policy-map WEBEX
Switch(config-pmap)#class WEBEX
Switch(config-pmap-c)#set dscp af41
Switch(config)#end

Switch(config)#interface fi4/0/5
Switch(config-if)#service-policy input WEBEX
Switch(config)#end

Switch#show run int fi4/0/5
Building configuration...

Current configuration : 98 bytes
!
interface FiveGigabitEthernet4/0/5
service-policy input WEBEX
ip nbar protocol-discovery
end
```

基於應用的Flexible NetFlow

有線AVC FNF支援兩種預定義流記錄：**舊式雙向流記錄**和新方**向流記錄**。

雙向流記錄用於跟蹤客戶端/伺服器應用程式統計資訊。

如輸出所示，雙向流記錄的示例配置。

```
Switch(config)#flow record BIDIR-1
Switch(config-flow-record)#match ipv4 version
Switch(config-flow-record)#match ipv4 protocol
Switch(config-flow-record)#match application name
Switch(config-flow-record)#match connection client ipv4 address
Switch(config-flow-record)#match connection server ipv4 address
Switch(config-flow-record)#match connection server transport port
Switch(config-flow-record)#match flow observation point
Switch(config-flow-record)#collect flow direction
Switch(config-flow-record)#collect connection initiator
Switch(config-flow-record)#collect connection new-connections
Switch(config-flow-record)#collect connection client counter packets long
Switch(config-flow-record)#connection client counter bytes network long
Switch(config-flow-record)#collect connection server counter packets long
Switch(config-flow-record)#connection server counter bytes network long
Switch(config-flow-record)#collect timestamp absolute first
Switch(config-flow-record)#collect timestamp absolute last
Switch(config-flow-record)#end

Switch#show flow record BIDIR-1
flow record BIDIR-1:
Description: User defined
No. of users: 0
Total field space: 78 bytes
Fields:
match ipv4 version
match ipv4 protocol
```

```
match application name
match connection client ipv4 address
match connection server ipv4 address
match connection server transport port
match flow observation point
collect flow direction
collect timestamp absolute first
collect timestamp absolute last
collect connection initiator
collect connection new-connections
collect connection server counter packets long
collect connection client counter packets long
collect connection server counter bytes network long
collect connection client counter bytes network long
```

方向記錄是用於輸入/輸出的應用統計資訊。

如輸出所示，輸入和輸出方向記錄的配置示例：

> **注意**：命令「**match interface input**」指定與輸入介面的匹配。命令「**match interface output**」指定與輸出介面匹配。命令「**match application name**」是AVC支援的必備命令。

```
Switch(config)#flow record APP-IN
Switch(config-flow-record)#match ipv4 version
Switch(config-flow-record)#match ipv4 protocol
Switch(config-flow-record)#match ipv4 source address
Switch(config-flow-record)#match ipv4 destination address
Switch(config-flow-record)#match transport source-port
Switch(config-flow-record)#match transport destination-port
Switch(config-flow-record)#match interface input
Switch(config-flow-record)#match application name
Switch(config-flow-record)#collect interface output
Switch(config-flow-record)#collect counter bytes long
Switch(config-flow-record)#collect counter packets long
Switch(config-flow-record)#collect timestamp absolute first
Switch(config-flow-record)#collect timestamp absolute last
Switch(config-flow-record)#end

Switch#show flow record APP-IN
flow record APP-IN:
Description: User defined
No. of users: 0
Total field space: 58 bytes
Fields:
match ipv4 version
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
match interface input
match application name
collect interface output
collect counter bytes long
collect counter packets long
collect timestamp absolute first
collect timestamp absolute last

Switch(config)#flow record APP-OUT
Switch(config-flow-record)#match ipv4 version
Switch(config-flow-record)#match ipv4 protocol
```

```
Switch(config-flow-record)#match ipv4 source address
Switch(config-flow-record)#match ipv4 destination address
Switch(config-flow-record)#match transport source-port
Switch(config-flow-record)#match transport destination-port
Switch(config-flow-record)#match interface output
Switch(config-flow-record)#match application name
Switch(config-flow-record)#collect interface input
Switch(config-flow-record)#collect counter bytes long
Switch(config-flow-record)#collect counter packets long
Switch(config-flow-record)#collect timestamp absolute first
Switch(config-flow-record)#collect timestamp absolute last
Switch(config-flow-record)#end

Switch#show flow record APP-OUT
flow record APP-OUT:
Description: User defined
No. of users: 0
Total field space: 58 bytes
Fields:
match ipv4 version
match ipv4 protocol
match ipv4 source address
match ipv4 destination address
match transport source-port
match transport destination-port
match interface output
match application name
collect interface input
collect counter bytes long
collect counter packets long
collect timestamp absolute first
collect timestamp absolute last
```

流匯出器

建立流匯出器以定義匯出引數。

如輸出所示，流匯出器的示例配置：

```
Switch(config)#flow exporter AVC
Switch(config-flow-exporter)#destination 192.168.69.2
Switch(config-flow-exporter)#source vlan69
Switch(config-flow-exporter)#end

Switch#show run flow exporter AVC
Current configuration:
!
flow exporter AVC
destination 192.168.69.2
source Vlan69
!
```

流量監控器

建立流監控器以將其與流記錄相關聯。

如輸出所示，流量監控器的示例配置：

```
Switch(config)#flow monitor AVC-MONITOR
```

```
Switch(config-flow-monitor)#record APP-OUT
Switch(config-flow-monitor)#exporter AVC
Switch(config-flow-monitor)#end

Switch#show run flow monitor AVC-MONITOR
Current configuration:
!
flow monitor AVC-MONITOR
exporter AVC
record APP-OUT
```

將流量監控器關聯到介面

最多可將具有不同預定義記錄的兩台不同的AVC監控器同時連線到一個介面。

如輸出所示，流量監控器的示例配置：

```
Switch(config)#interface fi4/0/5
Switch(config-if)#ip flow monitor AVC-MONITOR out
Switch(config-if)#end

Switch#show run interface fi4/0/5
Building configuration...
Current configuration : 134 bytes
!
interface FiveGigabitEthernet4/0/5
ip flow monitor AVC-MONITOR output
service-policy input WEBEX
ip nbar protocol-discovery
end
```

NBAR2

NBAR2動態無中斷協定包升級

通訊協定包是更新裝置上的NBAR2通訊協定支援（無需取代裝置上的思科軟體）的軟體套件。協定包包含由NBAR2正式支援的應用程式的資訊，這些應用程式經過編譯和打包在一起。對於每個應用程式，協定包都包含有關應用程式簽名和應用程式屬性的資訊。每個軟體版本都有一個內建的通訊協定包與之繫結。

- NBAR2提供了一種更新協定資料包的方法，無需中斷任何流量或服務，也無需修改裝置上的軟體映像
- NBAR2 protocol-packets可從以下URL下載到思科軟體中心：https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/qos_nbar/prot_lib/config_library/nbar-prot-pack-library.html

NBAR2協定包升級

在安裝新的協定包之前，必須將協定包複製到所有交換機的快閃記憶體中。 要載入新的協定包，請使用命令「ip nbar protocol-pack flash:<包名稱>」

無需重新載入交換機即可進行NBAR2升級。

如輸出所示，有關如何載入NBAR2協定包的示例配置：

```
Switch(config)#ip nbar protocol-pack flash:newProtocolPack
```

要恢復為內建協定包，請使用命令「default ip nbar protocol-pack」

如輸出所示，有關如何恢復到內建協定包的示例配置：

```
Switch(config)#default ip nbar protocol-pack
```

顯示NBAR2協定包資訊

要顯示協定包資訊，請使用下列命令：

- show ip nbar version
- show ip nbar protocol-pack active detail

如輸出所示，這些命令的示例輸出：

```
Switch#show ip nbar version
NBAR software version: 37
NBAR minimum backward compatible version: 37
NBAR change ID: 293126

Loaded Protocol Pack(s):
Name: Advanced Protocol Pack
Version: 43.0
Publisher: Cisco Systems Inc.
NBAR Engine Version: 37
State: Active

Switch#show ip nbar protocol-pack active detail
Active Protocol Pack:
Name: Advanced Protocol Pack
Version: 43.0
Publisher: Cisco Systems Inc.
NBAR Engine Version: 37
State: Active
```

NBAR2自定義應用程式

NBAR2支援使用自定義協定識別自定義應用程式。自定義協定支援NBAR2當前不支援的協定和應用程式。

它們可以包括：

- 組織的特定應用
- 特定地理位置的應用程式

NBAR2提供了一種通過ip nbar custom<myappname>命令手動自定義應用*程式的方法。*

> **附註**：自定義應用優先於內建協定

有多種型別的應用程式定製：

通用協定自定義

- HTTP
- SSL
- DNS

複合：基於多個協定的定製-server-name

第3層/第4層自定義

- IPv4地址
- DSCP值
- TCP/UDP埠
- 流源或目標方向

位元組偏移：基於負載中的特定位元組值的自定義

HTTP自定義

HTTP自定義可以基於來自以下位置的HTTP欄位組合：

- **cookie** - HTTP Cookie
- **host** — 包含資源的源伺服器的主機名
- **方法**- HTTP方法
- **referrer** — 從中獲取資源請求的地址
- **url** — 統一資源定位器路徑
- **user-agent** — 由傳送請求的代理使用的軟體
- **版本**- HTTP版本
- **via** - HTTP via欄位

名為MYHTTP的自定義應用程式示例，該應用程式使用具有選擇器ID 10的HTTP主機「*mydomain.com」。

```
Switch(config)#ip nbar custom MYHTTP http host *mydomain.com id 10
```

SSL自定義

通過從SSL伺服器名稱指示(SNI)或公用名稱(CN)提取的資訊，可以對SSL加密流量進行自定義。

名為MYSSL的自定義應用程式示例，該應用程式使用具有選擇器ID 11的SSL唯一名稱「mydomain.com」。

```
Switch(config)#ip nbar custom MYSSL ssl unique-name *mydomain.com id 11
```

DNS自定義

NBAR2會檢查DNS請求和響應流量，並會將DNS響應與應用程式相關聯。從DNS響應返回的IP地址將被快取並用於與該特定應用程式關聯的後續資料包流。

commandip nbar *customapplication-namednsdomain-nameidapplication-id*用於DNS自定義。要擴展應用程式，請使用commandip nbar *customapplication*-namedns domain-*namedomain-nameextendexisting-application*。

名為MYDNS的自定義應用程式示例，該應用程式使用具有選擇器ID 12的DNS域名「mydomain.com」。

```
Switch(config)#ip nbar custom MYDNS dns domain-name *mydomain.com id 12
```

複合定製

NBAR2提供了一種根據HTTP、SSL或DNS中顯示的域名自定義應用程式的方法。

名為MYDOMAIN的示例自定義應用程式，它使用選擇器ID為13的HTTP、SSL或DNS域名「mydomain.com」。

```
Switch(config)#ip nbar custom MYDOMAIN composite server-name *mydomain.com id 13
```

L3/L4自定義

第3層/第4層自定義基於資料包元組，並始終在流的第一個資料包上進行匹配。

將選擇器ID為14的IP地址10.56.1.10和10.56.1.11、TCP和DSCP ef進行匹配的自定義應用LAYER4CUSTOM示例。

```
Switch(config)#ip nbar custom LAYER4CUSTOM transport tcp id 14
Switch(config-custom)#ip address 10.56.1.10 10.56.1.11
Switch(config-custom)#dscp ef
Switch(config-custom)#end
```

監控自定義應用

要監控自定義應用程式，請使用列出的show命令：

show ip nbar protocol-id | inc自定義

```
Switch#show ip nbar protocol-id | inc Custom
LAYER4CUSTOM              14            Custom
MYDNS                     12            Custom
MYDOMAIN                  13            Custom
MYHTTP                    10            Custom
MYSSL                     11            Custom
```

show ip nbar protocol-id CUSTOM_APP

```
Switch#show ip nbar protocol-id MYSSL
Protocol Name              id            type
-----------------------------------------------
MYSSL                     11            Custom
```

驗證AVC

驗證AVC功能需要多個步驟，本節提供命令和示例輸出。

要驗證NBAR是否處於活動狀態，可以運行命令「**show ip nbar control-plane**」

關鍵領域：

- 在正確的情況下必須啟用**NBAR狀態**
- NBAR配置狀態必須**在正**確的場景中就緒

```
Switch#show ip nbar control-plane
NGCP Status:
============

graph sender info:
NBAR state is ACTIVATED
NBAR config send mode is ASYNC
NBAR config state is READY

NBAR update ID 3
NBAR batch ID ACK 3
NBAR last batch ID ACK clients 1 (ID: 4)
Active clients 1 (ID: 4)
NBAR max protocol ID ever 1935
NBAR Control-Plane Version: 37

<snip>
```

使用show platform software fed switch active|standby|member wdavc函式 wdavc_stile_cp_show_info_ui命令驗證每個交換機成員是否具有活動資料平面:

DP啟用後必須在正確的情況下為TRUE

```
Switch#show platform software fed switch active wdavc function wdavc_stile_cp_show_info_ui

Is DP activated : TRUE
MSG ID : 3
Maximum number of flows: 262144
Current number of graphs: 1
Requests queue state : WDAVC_STILE_REQ_QUEUE_STATE_UP
Number of requests in queue : 0
Max number of requests in queue (TBD): 1
Counters:
activate_msgs_rcvd : 1
graph_download_begin_msgs_rcvd : 3
stile_config_msgs_rcvd : 1584
graph_download_end_msgs_rcvd : 3
deactivate_msgs_rcvd : 0
intf_proto_disc_msgs_rcvd : 1
intf_attach_msgs_rcvd : 2
cfg_response_msgs_sent : 1593
num_of_handle_msg_from_fmanfp_events : 1594
num_of_handle_request_from_queue : 1594
num_of_handle_process_requests_events : 1594
```

使用「show platform software fed switch active|standby|member wdavc flows」命令顯示關鍵資訊：

```
Switch#show platform software fed switch active wdavc flows

CurrFlows=1, Watermark=1

IX |IP1 |IP2 |PORT1|PORT2|L3 |L4 |VRF |TIMEOUT|APP |TUPLE|FLOW |IS FIF |BYPASS|FINAL |#PKTS
|BYPASS
  | | | | |PROTO|PROTO|VLAN|SEC |NAME |TYPE |TYPE |SWAPPED | | | |PKT
--------------------------------------------------------------------------------
1 |192.168.100.2 |192.168.200.2 |68 |67 |1 |17 |0 |360 |unknown |Full |Real Flow|Yes |True |True
|40 |40
```

關鍵欄位：

CurrFlows：展示AVC跟蹤的活動流數量

水印:展示AVC歷史上跟蹤的最大流量數

超時秒:不活動超時取決於標識的應用程式

應用程式名稱：確定的應用程式

流型別:Real Flow表示此資料是因入站資料而建立的。預流表示此流由入站資料建立。預流用於預期的媒體流

元組型別:實際流始終為完整元組，預流為完整元組或半元組

旁路:如果設定為TRUE，則表示軟體不需要更多資料包來標識此流

最終:如果設定為TRUE，則表示此流不再更改應用程式

繞過PKT:達到最終分類需要多少資料包

#PKTS:此資料流實際上有多少資料包被傳送到軟體

檢視有關當前流的其他詳細資訊，可以使用命令「**show platform software fed switch active wdavc function wdavc_ft_show_all_flows_seg_ui**」

```
Switch#show platform software fed switch active wdavc function wdavc_ft_show_all_flows_seg_ui
CurrFlows=1, Watermark=1

IX |IP1 |IP2 |PORT1|PORT2|L3 |L4 |VRF |TIMEOUT|APP |TUPLE |FLOW |IS FIF |BYPASS|FINAL |#PKTS
|BYPASS
| | | | |PROTO|PROTO|VLAN|SEC |NAME |TYPE |TYPE |SWAPPED | | | |PKT
-------------------------------------------------------------------------------------------
-------------------------------------------------------------------
1 |192.168.100.2 |192.168.200.2|68 |67 |1 |17 |0 |360 |unknown |Full |Real Flow|Yes |True |True
|40 |40

SEG IDX |I/F ID |OPST I/F |SEG DIR |FIF DIR |Is SET |DOP ID |NFL HDL |BPS PND |APP PND |FRST TS
|LAST TS |BYTES |PKTS |TCP FLGS
-------------------------------------------------------------------------------------------
-----------------------------------
0 |9 |---- |Ingress |True |True |0 |50331823 |0 |0 |177403000|191422000|24252524|70094 |0
```

關鍵欄位

I/F ID:指定介面ID

SEG DIR：指定輸出方向的輸入

FIF DIR:確定這是否是流發起方方向

NFL HDL：硬體中的流ID

要檢視硬體中的條目，請運行命令「**show platform software fed switch active fnf flow-record asic**

<number> start-index <number> num-flows <number of flows>

附註：要選擇ASIC，它是埠對映到的ASIC例項。要標識ASIC，請使用「show platform software fed switch active|standby|member ifm mappings」命令。如果您對特定流不感興趣，可將啟動索引設定為「0」。否則，需要指定啟動索引。對於num-flows，指定可以檢視的流數，最多10個。

```
Switch#show platform software fed switch active fnf flow-record asic 3 start-index 0 num-flows 1
1 flows starting at 0 for asic 3:-------------------------------------------------
Idx 175 :
{90, ALR_INGRESS_NET_FLOW_ACL_LOOKUP_TYPE1 = 0x01}
{91, ALR_INGRESS_NET_FLOW_ACL_LOOKUP_TYPE2 = 0x01}
{0, ALR_INGRESS_NFL_SPECIAL1 = 0x00}
{11 PAD-UNK = 0x0000}
{94, PHF_INGRESS_DEST_PORT_OR_ICMP_OR_IGMP_OR_PIM_FIRST16B = 0x0043}
{93, PHF_INGRESS_SRC_PORT = 0x0044}
{67, PHF_INGRESS_IPV4_DEST_ADDRESS = 0xc0a8c802}
{68, PHF_INGRESS_IPV4_SRC_ADDRESS = 0xc0a86402}
{56, PHF_INGRESS_L3_PROTOCOL = 0x11}
FirstSeen = 0x2b4fb, LastSeen = 0x2eede, sysUptime = 0x2ef1c
PKT Count = 0x000000000001216f, L2ByteCount = 0x0000000001873006
```

在資料路徑中查詢各種錯誤和警告

使用「show platform software fed switch active|standby|member wdavc function wdavc_ft_show_stats_ui」命令 | inc err|warn|無法檢視可能的流表錯誤：

```
Switch#show platform software fed switch active wdavc function wdavc_ft_show_stats_ui | inc
err|warn|fail
Bucket linked exceed max error : 0
extract_tuple_non_first_fragment_warn : 0
ft_client_err_alloc_fail : 0
ft_client_err_detach_fail : 0
ft_client_err_detach_fail_intf_attach : 0
ft_inst_nfl_clock_sync_err : 0
ft_ager_err_invalid_timeout : 0
ft_intf_err_alloc_fail : 0
ft_intf_err_detach_fail : 0
ft_inst_err_unreg_client_all : 0
ft_inst_err_inst_del_fail : 0
ft_flow_seg_sync_nfl_resp_pend_del_warn : 0
ager_sm_cb_bad_status_err : 0
ager_sm_cb_received_err : 0
ft_ager_to_time_no_mask_err : 0
ft_ager_to_time_latest_zero_ts_warn : 0
ft_ager_to_time_seg_zero_ts_warn : 0
ft_ager_to_time_ts_bigger_curr_warn : 0
ft_ager_to_ad_nfl_resp_error : 0
ft_ager_to_ad_req_all_recv_error : 0
ft_ager_to_ad_req_error : 0
ft_ager_to_ad_resp_error : 0
ft_ager_to_ad_req_restart_timer_due_err : 0
ft_ager_to_flow_del_nfl_resp_error : 0
ft_ager_to_flow_del_all_recv_error : 0
ft_ager_to_flow_del_req_error : 0
ft_ager_to_flow_del_resp_error : 0
ft_consumer_timer_start_error : 0
ft_consumer_tw_stop_error : 0
```

```
ft_consumer_memory_error : 0
ft_consumer_ad_resp_error : 0
ft_consumer_ad_resp_fc_error : 0
ft_consumer_cb_err : 0
ft_consumer_ad_resp_zero_ts_warn : 0
ft_consumer_ad_resp_zero_pkts_bytes_warn : 0
ft_consumer_remove_on_count_zero_err : 0
ft_ext_field_ref_cnt_zero_warn : 0
ft_ext_gen_ref_cnt_zero_warn : 0
```

使用「show platform software fed switch active wdavc function wdavc_stile_stats_show_ui」命令 | inc err」檢視任何潛在的NBAR錯誤：

```
Switch#show platform software fed switch active wdavc function wdavc_stile_stats_show_ui | inc
err
find_flow_error : 0
add_flow_error : 0
remove_flow_error : 0
detach_fo_error : 0
is_forward_direction_error : 0
set_flow_aging_error : 0
ft_process_packet_error : 0
sys_meminfo_get_error : 0
```

驗證是否將資料包克隆到CPU

使用「show platform software fed switch active punt cpuq 21 | inc received"驗證是否將資料包克隆到CPU以進行NBAR處理：

> **附註**：在實驗室中，此數字沒有增加。

```
Switch#show platform software fed switch active punt cpuq 21 | inc received
Packets received from ASIC : 63
```

識別CPU擁塞

在擁塞時，封包可能會在傳送到WDAVC程式之前遭捨棄。使用show platform software fed switch active wdavc function fed_wdavc_show_ots_stats_ui命令進行驗證：

```
Switch#show platform software fed switch active wdavc function fed_wdavc_show_ots_stats_ui
OTS Limits
-------------------------------------------
ots_queue_max : 20000
emer_bypass_ots_queue_stress : 4000
emer_bypass_ots_queue_normal : 200
OTS Statistics
-------------------------------------------
total_requests : 40
total_non_wdavc_requests : 0
request_empty_field_data_error : 0
request_invalid_di_error : 0
request_buf_coalesce_error : 0
request_invalid_format_error : 0
request_ip_version_error : 0
request_empty_packet_error : 0
memory_allocation_error : 0
emergency_bypass_requests_warn : 0
dropped_requests : 0
```

```
enqueued_requests : 40
max_ots_queue : 0
```

> **提示：**要清除點投計數器，請使用「**show platform software fed switch active wdavc function fed_wdavc_clear_ots_stats_ui」命令**

確定擴展問題

如果硬體中沒有可用的FNF條目，則流量不屬於NBAR2分類。使用命令「**show platform software fed switch active fnf sw-table-sizes asic <number> shadow 0**」確認：

> **附註**：建立的流特定於交換機和asic核心。需要相應地指定交換器編號（主用、備用等）。輸入的ASIC編號繫結到各個介面，使用「**show platform software fed switch active|standby|member ifm mappings**」確定與該介面對應的ASIC。對於陰影選項，始終使用「0」。

```
Switch#show platform software fed switch active fnf sw-table-sizes asic 3 shadow 0

---------------------------------
Global Bank Allocation
---------------------------------
Ingress Banks : Bank 0
Egress Banks : Bank 1
---------------------------------
Global flow table Info
INGRESS usedBankEntry 1 usedOvfTcamEntry 0
EGRESS usedBankEntry 0 usedOvfTcamEntry 0 <-- 256 means TCAM entries are full
---------------------------------
Flows Statistics
INGRESS TotalSeen=1 MaxEntries=1 MaxOverflow=0
EGRESS TotalSeen=0 MaxEntries=0 MaxOverflow=0

---------------------------------
Partition Table
---------------------------------
## Dir Limit CurrFlowCount OverFlowCount MonitoringEnabled
0 ING 0 0 0 0
1 ING 16640 1 0 1
2 ING 0 0 0 0
3 ING 16640 0 0 0
4 ING 0 0 0 0
5 ING 8192 0 0 1
6 ING 0 0 0 0
7 ING 0 0 0 0
8 ING 0 0 0 0
9 ING 0 0 0 0
10 ING 0 0 0 0
11 ING 0 0 0 0
12 ING 0 0 0 0
13 ING 0 0 0 0
14 ING 0 0 0 0
15 ING 0 0 0 0
0 EGR 0 0 0 0
1 EGR 16640 0 0 1
2 EGR 0 0 0 0
3 EGR 16640 0 0 0
4 EGR 0 0 0 0
5 EGR 8192 0 0 1
6 EGR 0 0 0 0
```

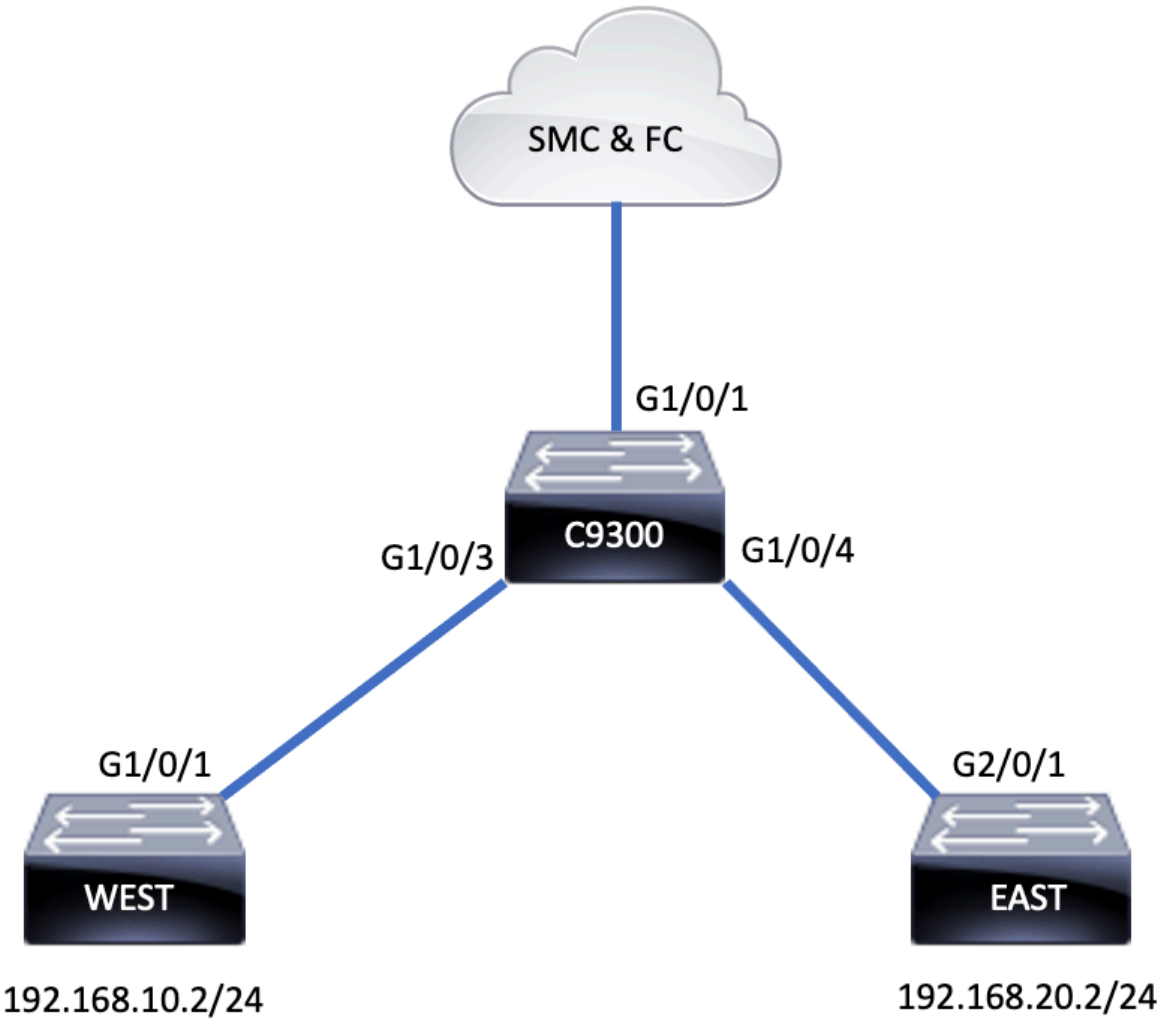
```
7 EGR 0 0 0 0
8 EGR 0 0 0 0
9 EGR 0 0 0 0
10 EGR 0 0 0 0
11 EGR 0 0 0 0
12 EGR 0 0 0 0
13 EGR 0 0 0 0
14 EGR 0 0 0 0
15 EGR 0 0 0 0
```

加密流量分析(ETA)

背景資訊

- ETA專注於通過被動監控、提取相關資料元素、將行為建模和機器學習與基於雲的全球安全相結合，識別加密流量中的惡意軟體通訊。
- ETA利用來自NetFlow的遙測以及加密的惡意軟體檢測和加密合規性，並將此資料傳送到Cisco Stealthwatch。
- ETA提取兩個主要的資料元素：初始資料包(IDP)和資料包長度和時間序列(SPLT)。

網路圖表



元件

ETA由多個不同的元件組成，這些元件用於建立ETA解決方案：

- NetFlow — 一種標準，用於定義網路裝置匯出的資料元素，這些元素描述網路上的流量。
- Cisco Stealthwatch — 利用網路遙測功能（包括NetFlow、IPFIX、代理日誌以及對原始資料包的深度資料包檢測），提供高級網路可視性、安全情報和分析。
- 思科認知智慧 — 查詢繞過安全控制或通過未受監控的管道進入組織環境內的惡意活動。
- 加密流量分析 — Cisco IOS XE功能使用高級行為演算法通過分析加密流量的傳入後設資料來識別惡意流量模式，可檢測加密流量中隱藏的潛在威脅。

附註：本文此部分僅重點介紹在Catalyst 9000系列交換機上配置和驗證ETA和NetFlow，但不涵蓋部署到Cognitive Intelligence Cloud的Stealthwatch管理控制檯(SMC)和流量收集器(FC)。

限制

- 部署ETA需要DNA優勢發揮作用
- 同一介面不支援ETA和傳輸(TX)交換連線埠分析器(SPAN)。

這不是一個包含清單，請查閱交換機的相應配置指南和所有限制的代碼版本。

組態

如輸出所示，在交換機上全域性啟用ETA並定義流匯出目標：

```
C9300(config)#et-analytics
C9300(config-et-analytics)#ip flow-export destination 172.16.18.1 2055
```

提示：必須使用埠2055，不要使用其他埠號。

接下來，配置Flexible NetFlow，如輸出所示：

配置流記錄

```
C9300(config)#flow record FNF-RECORD
C9300(config-flow-record)#match ipv4 protocol
C9300(config-flow-record)#match ipv4 source address
C9300(config-flow-record)#match ipv4 destination address
C9300(config-flow-record)#match transport source-port
C9300(config-flow-record)#match transport destination-port
C9300(config-flow-record)#collect counter bytes long
C9300(config-flow-record)#collect counter packets long
C9300(config-flow-record)#collect timestamp absolute first
C9300(config-flow-record)#collect timestamp absolute last
```

配置流監控

```
C9300(config)#flow exporter FNF-EXPORTER
C9300(config-flow-exporter)#destination 172.16.18.1
C9300(config-flow-exporter)#transport udp 2055
C9300(config-flow-exporter)#template data timeout 30
```

```
C9300(config-flow-exporter)#option interface-table
C9300(config-flow-exporter)#option application-table timeout 10
C9300(config-flow-exporter)#exit
```

配置流記錄

```
C9300(config)#flow monitor FNF-MONITOR
C9300(config-flow-monitor)#exporter FNF-EXPORTER
C9300(config-flow-monitor)#record FNF-RECORD
C9300(config-flow-monitor)#end
```

應用流監視器

```
C9300(config)#int range g1/0/3-4
C9300(config-if-range)#ip flow mon FNF-MONITOR in
C9300(config-if-range)#ip flow mon FNF-MONITOR out
C9300(config-if-range)#end
```

在交換器介面上啟用ETA

```
C9300(config)#interface range g1/0/3-4
C9300(config-if-range)#et-analytics enable
```

驗證

驗證ETA監控器「eta-mon」監控器是否處於活動狀態。通過**show flow monitor eta-mon**命令確認狀態已分配

```
C9300#show flow monitor eta-mon
Flow Monitor eta-mon:
Description: User defined
Flow Record: eta-rec
Flow Exporter: eta-exp
Cache:
Type: normal (Platform cache)
Status: allocated
Size: 10000 entries
Inactive Timeout: 15 secs
Active Timeout: 1800 secs
```

驗證ETA快取已填充。當在同一介面上配置NetFlow和ETA時，使用「**show flow monitor <monitor name> cache**」而不是「**show flow monitor eta-mon cache**」，因為「**show flow monitor eta-mon cache」的輸出為空：**

```
C9300#show flow monitor FNF-MONITOR cache
Cache type: Normal (Platform cache)
Cache size: 10000
Current entries: 4

Flows added: 8
Flows aged: 4
- Inactive timeout ( 15 secs) 4

IPV4 SOURCE ADDRESS: 192.168.10.2
IPV4 DESTINATION ADDRESS: 192.168.20.2
TRNS SOURCE PORT: 0
TRNS DESTINATION PORT: 0
IP PROTOCOL: 1
```

```
counter bytes long: 500
counter packets long: 5
timestamp abs first: 21:53:23.390
timestamp abs last: 21:53:23.390

IPV4 SOURCE ADDRESS: 192.168.20.2
IPV4 DESTINATION ADDRESS: 192.168.10.2
TRNS SOURCE PORT: 0
TRNS DESTINATION PORT: 0
IP PROTOCOL: 1
counter bytes long: 500
counter packets long: 5
timestamp abs first: 21:53:23.390
timestamp abs last: 21:53:23.390

IPV4 SOURCE ADDRESS: 192.168.20.2
IPV4 DESTINATION ADDRESS: 192.168.10.2
TRNS SOURCE PORT: 0
TRNS DESTINATION PORT: 0
IP PROTOCOL: 1
counter bytes long: 500
counter packets long: 5
timestamp abs first: 21:53:23.390
timestamp abs last: 21:53:23.390

IPV4 SOURCE ADDRESS: 192.168.10.2
IPV4 DESTINATION ADDRESS: 192.168.20.2
TRNS SOURCE PORT: 0
TRNS DESTINATION PORT: 0
IP PROTOCOL: 1
counter bytes long: 500
counter packets long: 5
timestamp abs first: 21:53:23.390
timestamp abs last: 21:53:23.390
```

使用「show flow exporter eta-exp statistics」命令驗證流是否匯出到SMC和FC。

```
C9300#show flow exporter eta-exp statistics
Flow Exporter eta-exp:
Packet send statistics (last cleared 03:05:32 ago):
Successfully sent: 3 (3266 bytes)

Client send statistics:
Client: Flow Monitor eta-mon
Records added: 4
- sent: 4
Bytes added: 3266
- sent: 3266
```

使用「show platform software fed switch active fnf et-analytics-flows」命令確認SPLT和IDP已匯出到FC。

```
C9300#show platform software fed switch active fnf et-analytics-flows

ET Analytics Flow dump

=================
Total packets received : 20
Excess packets received : 0
Excess syn received : 0
Total eta records added : 4
Current eta records : 0
```

```
Total eta splt exported : 2
Total eta IDP exported : 2
```

使用「show platform software et-analytics interfaces」命令驗證為et-analytics配置了哪些接口

```
C9300#show platform software et-analytics interfaces
ET-Analytics interfaces
GigabitEthernet1/0/3
GigabitEthernet1/0/4

ET-Analytics VLANs
```

使用命令「show platform software et-analytics global」檢視ETA的全域性狀態：

```
C9300#show plat soft et-analytics global
ET-Analytics Global state
=========================
All Interfaces : Off
IP Flow-record Destination : 10.31.126.233 : 2055
Inactive timer : 15

ET-Analytics interfaces
GigabitEthernet1/0/3
GigabitEthernet1/0/4

ET-Analytics VLANs
```

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）。