

Configuración y verificación de Netflow, AVC y ETA en switches Catalyst serie 9000

Contenido

Introducción

Este documento describe cómo configurar y validar NetFlow, Application Visibility and Control (AVC) y Encrypted Traffic Analytics (ETA).

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Netflow
- AVC
- ETA

Componentes Utilizados

La información de este documento se basa en un switch Catalyst 9300 que ejecuta el software Cisco IOS® XE 16.12.4.

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Productos Relacionados

Este documento también puede utilizarse con estas versiones de software y hardware:

- 9200
- 9400
- 9500
- 9600
- Cisco IOS XE 16.12 y posterior

Antecedentes

- Flexible NetFlow es la tecnología de flujo de última generación que recopila y mide datos para permitir que todos los routers o switches de la red se conviertan en una fuente de telemetría.
- Flexible NetFlow permite realizar mediciones del tráfico extremadamente granulares y precisas, así como recopilar tráfico agregado de alto nivel.
- Flexible NetFlow utiliza flujos para proporcionar estadísticas para la contabilidad, la supervisión de la red y la planificación de la red.
- Un flujo es un flujo unidireccional de paquetes que llega a una interfaz de origen y tiene los mismos valores para las claves. Una clave es un valor identificado para un campo dentro del paquete. Puede crear un flujo a través de un registro de flujo para definir las claves únicas para el flujo.


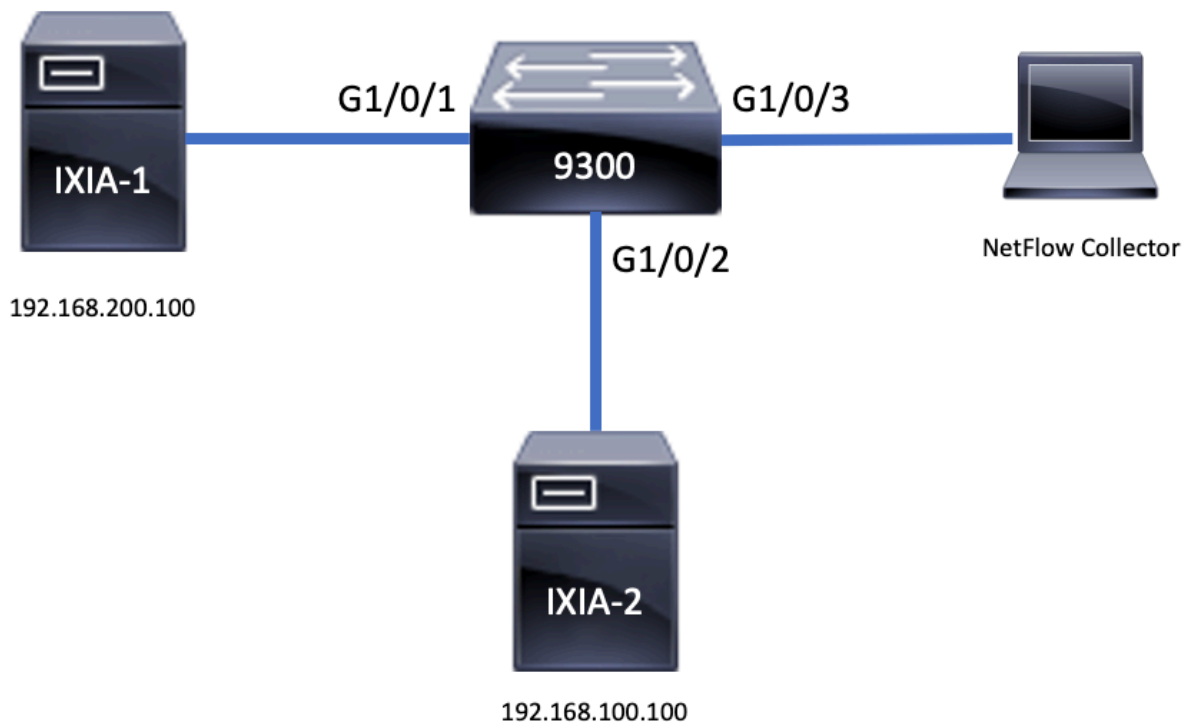
 Nota: Los comandos de plataforma (fed) pueden variar. El comando puede ser `show platform fed` versus `show platform fed switch`. Si la sintaxis anotada en los ejemplos no se analiza, por favor intente la variante.

Diagrama de la red



Configurar

Componentes

La configuración de NetFlow se compone de tres componentes principales que se pueden utilizar juntos, con varias variaciones para realizar análisis de tráfico y exportación de datos.

Registro de Flujo

- Un registro es una combinación de campos clave y no clave. Los registros de Flexible NetFlow se asignan a los monitores de flujo de Flexible NetFlow para definir la caché que se utiliza para el almacenamiento de los datos de flujo.
- Flexible NetFlow incluye varios registros predefinidos que se pueden utilizar para supervisar el tráfico.
- Flexible NetFlow también permite definir registros personalizados para una caché de monitor de flujo de Flexible NetFlow mediante la especificación de campos clave y no clave para personalizar la recopilación de datos según sus requisitos específicos.

Como se muestra en el ejemplo, los detalles de configuración del registro de flujo:

```
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
```

Exportador de flujo

- Los exportadores de flujo se utilizan para exportar los datos de la caché de monitor de flujo a un sistema remoto (servidor que funciona como recopilador de NetFlow), para su análisis y almacenamiento.
- Los exportadores de flujo se asignan a los monitores de flujo para proporcionar la capacidad de exportación de datos para los monitores de flujo.

Como se muestra en el ejemplo, los detalles de configuración del exportador de flujo:

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

Monitor de Flujo

- Los monitores de flujo son el componente de Flexible NetFlow que se aplica a las interfaces para realizar la supervisión del tráfico de red.
- Los datos de flujo se recopilan del tráfico de red y se agregan a la caché de supervisión de flujo mientras se ejecuta el proceso. El proceso se basa en los campos clave y no clave del registro de flujo.

Como se muestra en el ejemplo, los detalles de configuración del monitor de flujo:

```
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
```

Muestreador de flujo (opcional)

- Los muestreadores de flujo se crean como componentes independientes en la configuración de un router.
- Los muestreadores de flujo limitan el número de paquetes que se seleccionan para el análisis a fin de reducir la carga en el dispositivo que utiliza Flexible NetFlow.
- Los muestreadores de flujo se utilizan para reducir la carga en el dispositivo que utiliza Flexible NetFlow alcanzada a través del límite del número de paquetes seleccionados para el análisis.
- Los muestreadores de flujo intercambian precisión para el rendimiento del router. Si hay una reducción en el número de paquetes que son analizados por el monitor de flujo, la precisión de la información almacenada en la memoria caché del monitor de flujo puede verse afectada.

Como se muestra en el ejemplo, configuración del muestreador de flujo de ejemplo:

```
<#root>
```

```
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
```

Restricciones

- La licencia DNA Addon es necesaria para Flexible NetFlow completo; de lo contrario, Sampled NetFlow sólo está disponible.
- Los exportadores de flujo no pueden utilizar el puerto de administración como origen.

Esta no es una lista inclusiva, consulte la guía de configuración de la plataforma y el código apropiados.

Verificación

Verificación independiente de la plataforma

Verifique la configuración y confirme que los componentes de NetFlow necesarios están presentes:

1. Registro de Flujo
2. Exportador de flujo
3. Monitor de Flujo
4. Muestreador de flujo (opcional)



Sugerencia: para ver el registro de flujo, el exportador de flujo y la salida del monitor de flujo en un comando, ejecute `show running-config flow monitor`

expand

Como se muestra en el ejemplo, el monitor de flujo está vinculado a la dirección de entrada y sus componentes asociados:

<#root>

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  
!
```

Como se muestra en el ejemplo, el monitor de flujo está vinculado a la dirección de salida y sus componentes asociados:

<#root>

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  
!
```

Ejecute el comando `show flow monitor`

```
statistics
```

. Este resultado es útil para confirmar que se registran los datos:

```
<#root>
```

```
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
```

Ejecute el comando `show flow monitor`

```
cache
```

para confirmar que la memoria caché de NetFlow tiene resultados:

```
<#root>
```

```
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
```

Ejecute el comando `show flow exporter`

```
statistics
```

para confirmar que el exportador envió paquetes:

```
<#root>
```

```
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
```

Verificación dependiente de la plataforma

Inicialización de NetFlow - Tabla de particiones NFL

- Las particiones NetFlow se inicializan para diferentes funciones con 16 particiones por dirección (Entrada vs Salida).
- La configuración de la tabla de particiones de NetFlow se divide en la asignación de banco global, que se subdivide en los bancos de flujo de entrada y salida.

Campos Llave


- Número de particiones
- Estado de activación de partición
- Límite de partición
- Uso actual de la partición

Para ver la tabla de particiones de NetFlow, puede ejecutar el comando `show platform software fed switch`

```
active|standby|member| fnf sw-table-sizes asic
```

```
shadow 0
```

```
.
```


 Nota: Los flujos que se crean son específicos del switch y del núcleo básico cuando se crean. El número de switch (activo, en espera, etc.) debe especificarse en consecuencia. El número ASIC que se ingresa está vinculado a la interfaz respectiva, utilice `show platform software fed switch active|standby|member ifm mappings` para determinar el ASIC que corresponde a la interfaz. Para la opción de sombra, utilice siempre "0".

<#root>

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 used0vfTcamEntry    0  
EGRESS    usedBankEntry      0 used0vfTcamEntry    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

Monitor de Flujo

La configuración del monitor de flujo incluye lo siguiente:

1. Configuración de ACL de NetFlow, que da como resultado la creación de una entrada dentro de la tabla TCAM de ACL.

La entrada TCAM de ACL está compuesta por:


- Buscar claves coincidentes
- Parámetros de resultado utilizados para la búsqueda de NetFlow, que incluye lo siguiente:
 - ID de perfil
 - ID de NetFlow

2. Configuración de Máscara de Flujo, que da como resultado la creación de una entrada en NflLookupTable y NflFlowMaskTable.

- Indexado por parámetros de resultado de ACL de NetFlow para encontrar la máscara de flujo para la búsqueda de NetFlow

ACL de NetFlow

Para ver la configuración de la ACL de NetFlow, ejecute el comando `show platform hardware fed switch active fwd-asic resource tcam table nfl_acl asic`

 Sugerencia: si hay una ACL de puerto (PACL), la entrada se crea en el ASIC al que está asignada la interfaz. En el caso de una ACL de router (RACL), la entrada está presente en todos los ASIC.

- En esta salida hay NFCMD0 y NFCMD1, que son valores de 4 bits. Para calcular el ID de

perfil, convierta los valores en binarios.

- En esta salida, NFCMD0 es 1, NFCMD1 es 2. Cuando se convierte en binario: 000100010
- En Cisco IOS XE 16.12 y versiones posteriores dentro de los 8 bits combinados, los primeros 4 bits son el ID de perfil y el séptimo bit indica que la búsqueda está habilitada. En el ejemplo, 00010010, el ID de perfil es 1.
- En Cisco IOS XE 16.11 y versiones anteriores de código, dentro de los 8 bits combinados, los primeros 6 bits son el ID de perfil, y el 7º bit indica que la búsqueda está habilitada. En este ejemplo, 00010010, el ID de perfil es 4.

<#root>

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	ME	IP	OPT	MF	NFF	DF	SO	DPT	TM	DSE	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	DstPort	IITypeCode	TCPF	Flags	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	AuthBehavior	Tag	l2srcMiss	l2dstMiss	ipTtl	SgacIDeny
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	LUTOPRI	CPUCOPY
1	2	0	1	0	0	0	0	0	0x0000E	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	DstPort	IITypeCode	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	AuthBehavior	Tag	l2srcMiss	l2dstMiss	ipTtl	SgacIDeny
M:	0	000000	0	0	0	0	0	0
V:	0	000000	0	0	0	0	0	0

NFCMD0	NFCMD1	SMPLR	LKP1	LKP2	PID	QOSPRI	MLLBL	MPLPRO	LUTOPRI	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	DstPort	IITypeCode	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	AuthBehavior	Tag	l2srcMiss	l2dstMiss	ipTtl	SgacIDeny
M:	0	000000	0	0	0	0	0	0
V:	0	000000	0	0	0	0	0	0

NFCMD0	NFCMD1	SMPLR	LKP1	LKP2	PID	QOSPRI	MLLBL	MPLPRO	LUTOPRI	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 T3Len fLabel vrfId toUs
00000000 00000000 00000000 00 00 0000 00000 000 0
00000000 00000000 00000000 00 00 0000 00000 000 0

T3Pro mtrId AE FE RE HE MF NFF SO IPOPT RA MEn RMAC DPT TMP T3m
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 dstPort IITypeCode 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 SMLPR LKP1 LKP2 PID QOSPRI MQLBL MPLPRO LUTOPRI CPUCOPY
0 0 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 SMLPR LKP1 LKP2 PID QOSPRI MQLBL MPLPRO LUTOPRI CPUCOPY
0 0 0 0 0 0 0 0 0 0x00000 0
Start/Skip Word: 0x00000000
No Start, Terminate

```

Máscara de flujo

Ejecute el comando `show platform software fed switch active|standby|member fnf fmask-entry asic`

```
entry 1
```

para ver que la máscara de flujo está instalada en el hardware. El número de la lista de campos clave también se puede encontrar aquí.

```
<#root>
```

```
Switch#
```

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

```

-----
mask0_valid : 1
Mask hd10   : 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)

Estadísticas de flujo y datos de descarga de marca de tiempo

Ejecute el comando `show platform software fed switch active fnf flow-record asic`

```
start-index
```

```
num-flows
```

para ver las estadísticas de NetFlow así como las marcas de tiempo

```
<#root>
```

```
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 = 0x00000000102d5df, 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 = 0x000000001050682, L2ByteCount = 0x00000000cbcd1590
```

Visibilidad y control de aplicaciones (AVC)

Antecedentes

- Application Visibility and Control (AVC) es una solución que aprovecha Network-Based Recognition Version 2 (NBAR2), NetFlow V9 y diversas herramientas de informes y gestión

(Cisco Prime) para ayudar a clasificar aplicaciones mediante inspección profunda de paquetes (DPI).

- AVC se puede configurar en puertos de acceso por cable para switches independientes o pilas de switches.
- AVC también se puede utilizar en los controladores inalámbricos de Cisco para identificar aplicaciones basadas en PPP y, a continuación, marcarlas con un valor DSCP específico. También puede recopilar diversas métricas de rendimiento inalámbrico, como el uso de ancho de banda en términos de aplicaciones y clientes.

Rendimiento y escalabilidad

Rendimiento: cada miembro del switch puede gestionar 500 conexiones por segundo (CPS) con una utilización de la CPU inferior al 50%. Más allá de esta tarifa, el servicio AVC no está garantizado.

Escalabilidad: capacidad para gestionar hasta 5000 flujos bidireccionales por cada 24 puertos de acceso (aproximadamente 200 flujos por puerto de acceso).

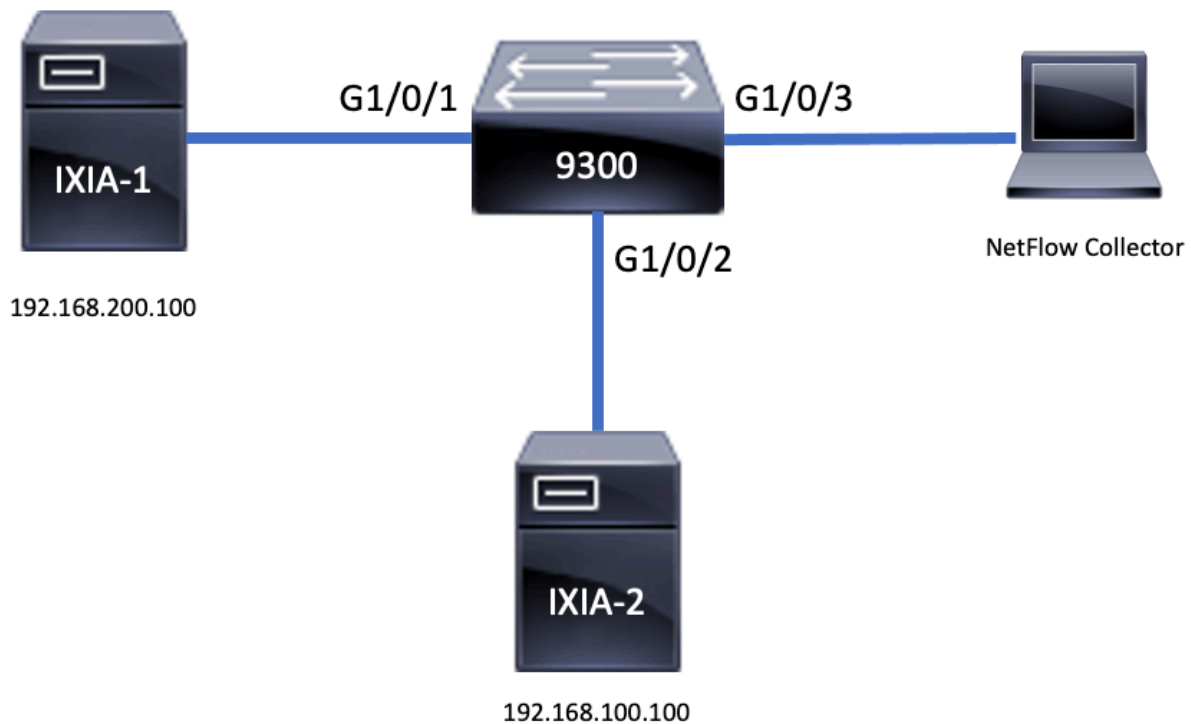
Restricciones de AVC por cable

- AVC y Encrypted Traffic Analytics (ETA) no se pueden configurar juntos al mismo tiempo en la misma interfaz.
- La clasificación de paquetes sólo se admite para el tráfico de unidifusión IPv4 (TCP/UDP).
- La configuración de políticas de QoS basada en NBAR solo se admite en puertos físicos con cables. Esto incluye los puertos troncales y de acceso de capa 2 y los puertos enrutados de capa 3.
- La configuración de políticas de QoS basada en NBAR no se admite en miembros de canal de puerto, interfaces virtuales de switch (SVI) o subinterfaces.
- Los clasificadores basados en NBAR2 (protocolo de coincidencia) sólo admiten acciones de QoS de marcación y regulación.
- El "protocolo de coincidencia" está limitado a 255 protocolos diferentes en todas las políticas (limitación de hardware de 8 bits)



Nota: Esta no es una lista exhaustiva de todas las restricciones, consulte la guía de configuración de AVC adecuada para su plataforma y versión de código.

Diagrama de la red



Componentes

La configuración de AVC consta de tres componentes principales que conforman la solución:

Visibilidad: Detección de protocolos

- La detección de protocolos se logra a través de NBAR, que proporciona estadísticas de bytes/paquetes por interfaz, dirección y aplicación.
- La detección de protocolos se habilita para una interfaz específica a través de la configuración de la interfaz: `ip nbar protocol-discovery`.

Como se muestra en el resultado, cómo habilitar la detección de protocolos:

```
<#root>
```

```
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
```

Control: QoS basada en aplicaciones

En comparación con la QoS tradicional que coincide en la dirección IP y el puerto UDP/TCP, AVC logra un control más preciso a través de QoS basada en aplicaciones, que permite la coincidencia en la aplicación, y proporciona un control más granular a través de acciones de QoS como la marcación y la regulación.

- Las acciones se realizan en el tráfico agregado (no por flujo).
- La QoS basada en aplicaciones se logra mediante la creación de un mapa de clase, la coincidencia de un protocolo y, a continuación, la creación de un mapa de política.
- La política de QoS basada en la aplicación se asocia a una interfaz.

Como se muestra en el resultado, ejemplo de configuración para QoS basada en aplicación:

```
<#root>
```

```
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
```

Flexible NetFlow basado en aplicaciones

El FNF de AVC por cable admite dos tipos de registros de flujo predefinidos: los registros de flujo bidireccionales heredados y los nuevos registros de flujo direccional.

Los registros de flujo bidireccionales realizan un seguimiento de las estadísticas de las aplicaciones cliente/servidor.

Como se muestra en la salida, ejemplo de configuración de un registro de flujo bidireccional.

```
<#root>
```

```
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
```

Los registros direccionales son estadísticas de aplicación para entrada/salida.

Como se muestra en la salida, ejemplos de configuración de registros direccionales de entrada y salida:

 Nota: El comando `match interface input` especifica una coincidencia con la interfaz de entrada. El comando `match interface output` especifica una coincidencia con la interfaz de salida. El comando `match application name` es obligatorio para el soporte de AVC.

<#root>

```
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
```

Exportador de flujo

Cree un exportador de flujo para definir los parámetros de exportación.

Como se muestra en el resultado, ejemplo de configuración del exportador de flujo:

```
<#root>  
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  
!
```

Monitor de Flujo

Cree un monitor de flujo para asociarlo a un registro de flujo.

Como se muestra en la salida, configuración de ejemplo del monitor de flujo:

```
<#root>
```

```
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
```

Asociación de un Monitor de Flujo a una Interfaz

Puede conectar hasta dos monitores AVC diferentes con diferentes registros predefinidos a una interfaz al mismo tiempo.

Como se muestra en la salida, configuración de ejemplo del monitor de flujo:

```
<#root>
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

Actualización del paquete de protocolos dinámicos sin impacto NBAR2

Los paquetes de protocolo son paquetes de software que actualizan la compatibilidad con el protocolo NBAR2 en un dispositivo sin reemplazar el software de Cisco en el dispositivo. Un paquete de protocolo contiene información sobre las aplicaciones oficialmente admitidas por NBAR2 que se compilan y empaquetan juntas. Para cada aplicación, el paquete de protocolos incluye información sobre las firmas y los atributos de la aplicación. Cada versión de software

incluye un paquete de protocolos integrado.

- NBAR2 proporciona una forma de actualizar el paquete de protocolo sin ninguna interrupción del tráfico o del servicio y sin necesidad de modificar la imagen de software en los dispositivos.
- Los paquetes de protocolo NBAR2 están disponibles para su descarga en Cisco Software Center en: [Biblioteca de paquetes de protocolos NBAR2](#) .

Actualización del paquete de protocolos NBAR2

Antes de instalar un nuevo paquete de protocolo, debe copiar el paquete de protocolo en la memoria flash en todos los switches. Para cargar el nuevo paquete de protocolo, utilice el comando `ip nbar protocol-pack flash:`

No es necesario volver a cargar los switches para que se produzca la actualización de NBAR2.

Como se muestra en el resultado, configuración de ejemplo de cómo cargar el paquete de protocolo NBAR2:

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

Para volver al paquete de protocolos integrado, utilice el comando `default ip nbar protocol-pack`.

Como se muestra en el resultado, configuración de ejemplo de cómo volver al paquete de protocolo integrado:

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

Mostrar información del paquete del protocolo NBAR2

Para visualizar la información del paquete de protocolos, utilice los comandos enumerados:

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

Como se muestra en el resultado, ejemplo de resultado de esos comandos:

```
<#root>
```

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

Aplicaciones personalizadas de NBAR2

NBAR2 admite el uso de protocolos personalizados para identificar las aplicaciones personalizadas. Los protocolos personalizados admiten protocolos y aplicaciones que NBAR2 no admite actualmente.

Estos pueden incluir lo siguiente:

- Aplicación específica a una organización
- Aplicaciones específicas de una zona geográfica

NBAR2 proporciona una forma de personalizar manualmente las aplicaciones mediante el comando `ip nbar custom`



Nota: las aplicaciones personalizadas tienen prioridad sobre los protocolos integrados

Existen varios tipos de personalización de aplicaciones:

Personalización de protocolo genérico

- HTTP
- SSL
- DNS

Compuesto:personalización basada en varios protocolos -nombre de servidor.

Personalización de capa 3/capa 4

- Dirección IPv4
- Valores DSCP
- Puertos TCP/UDP
- Dirección de origen o destino del flujo

Desplazamiento de bytes:personalización basada en valores de bytes específicos de la carga útil

Personalización de HTTP

La personalización de HTTP podría basarse en una combinación de campos HTTP de:

- cookie - Cookie HTTP
- host: nombre de host de Origin Server que contiene el recurso
- method - Método HTTP
- referrer - Dirección de la cual se obtuvo la solicitud de recurso
- url - Ruta del Localizador Uniforme de Recursos
- user-agent - Software utilizado por el agente que envía la solicitud
- versión: versión HTTP
- vía: campo HTTP vía

Ejemplo de aplicación personalizada denominada MYHTTP que utiliza el host HTTP *mydomain.com con ID de selector 10.

```
<#root>
```

```
Switch(config)#
```

```
ip nbar custom MYHTTP http host *mydomain.com id 10
```

Personalización de SSL

La personalización se puede realizar para el tráfico cifrado SSL a través de la información extraída de la indicación de nombre de servidor SSL (SNI) o del nombre común (CN).

Ejemplo de aplicación personalizada denominada MYSSL que utiliza SSL unique-name mydomain.com con identificador de selector 11.

```
<#root>
```

```
Switch(config)#
```

```
ip nbar custom MYSSL ssl unique-name *mydomain.com id 11
```

Personalización de DNS

NBAR2 examina el tráfico de solicitudes y respuestas DNS y puede correlacionar la respuesta DNS con una aplicación. La dirección IP devuelta por la respuesta DNS se almacena en caché y se utiliza para flujos de paquetes posteriores asociados a esa aplicación específica.

El comando `ip nbar custom application-namedns domain-nameidapplication-id` se utiliza para la personalización de DNS. Para extender una aplicación, utilice el comando `ip nbar custom application-name dns domain-name domain-name extends existing-application`.

Aplicación personalizada de ejemplo denominada MYDNS que utiliza el nombre de dominio DNS "mydomain.com" con ID de selector 12.

```
<#root>
```

```
Switch(config)#
```

```
ip nbar custom MYDNS dns domain-name *mydomain.com id 12
```

Personalización compuesta

NBAR2 proporciona una forma de personalizar aplicaciones basadas en nombres de dominio que aparecen en HTTP, SSL o DNS.

Ejemplo de aplicación personalizada denominada MYDOMAIN que utiliza el nombre de dominio HTTP, SSL o DNS mydomain.com con ID de selector 13.

```
<#root>
```

```
Switch(config)#
```

```
ip nbar custom MYDOMAIN composite server-name *mydomain.com id 13
```

Personalización L3/L4

La personalización de la capa 3/capa 4 se basa en la tupla de paquetes y siempre coincide en el primer paquete de un flujo.

Ejemplo de aplicación personalizada LAYER4CUSTOM que coincide con las direcciones IP 10.56.1.10 y 10.56.1.11, TCP y DSCP ef con el ID de selector 14.

```

<#root>
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

```

Supervisar aplicaciones personalizadas

Para supervisar las aplicaciones personalizadas, utilice los comandos show enumerados:

```
show ip nbar protocol-id | inc personalizado
```

```

<#root>
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
```

```

<#root>
Switch#
show ip nbar protocol-id MYSSL

Protocol Name          id          type
-----
MYSSL                  11          Custom

```

Verificar AVC

Existen varios pasos para validar la funcionalidad de AVC; en esta sección se proporcionan comandos y ejemplos de resultados.

Para validar que NBAR está activo, puede ejecutar el comando `show ip nbar control-plane`.

Áreas clave:

- El estado NBAR debe estar activado en un escenario correcto
- El estado de configuración de NBAR debe estar listo en un escenario correcto

<#root>

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>

Valide que cada miembro del switch tenga un plano de datos activo con el comando `show platform`

`software fed switch active|standby|member wdacv function wdacv_stile_cp_show_info_ui`:

El valor de DP activado debe ser TRUE en un escenario correcto

<#root>

Switch#

```
show platform software fed switch active wdacv function wdacv_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

Utilice el comando `show platform software fed switch active|standby|member wдавc flows` para mostrar información clave:

<#root>

Switch#

`show platform software fed switch active wдавc flows`

CurrFlows=1, Watermark=1

IX	IP1	IP2	PORT1	PORT2	L3	L4	VRF	TIMEOUT	APP	TUPLE	FLOW	IS FIF
					PROTO	PROTO	VLAN	SEC	NAME	TYPE	TYPE	SWAPPED
1	192.168.100.2	192.168.200.2	68	67	1	17	0	360	unknown	Full	Real Flow	Yes

Campos Llave:

CurrFlows: muestra cuántos flujos activos controla AVC

Marca de agua: muestra el mayor número de flujos históricamente rastreados por AVC

TIMEOUT SEC: tiempo de espera de inactividad basado en la aplicación identificada

APP NAME: Aplicación identificada

TIPO DE FLUJO: El flujo real indica que se creó como resultado de datos entrantes. Pre Flow indica que este flujo se crea como resultado de datos entrantes. Los flujos previos se utilizan para los flujos de medios previstos

TIPO DE TUPLA: Los flujos reales son siempre tupla completa, los flujos previos son tupla completa o media tupla

BYPASS: Si se establece en TRUE, indica que el software no necesita más paquetes para identificar este flujo

FINAL: si se establece en TRUE, indica que la aplicación ya no cambia para este flujo

BYPASS PKT: Cantidad de paquetes necesarios para llegar a la clasificación final

#PKTS: ¿Cuántos paquetes se han enviado realmente al software para este flujo?

Ver detalles adicionales sobre los flujos actuales, puede utilizar el comando `show platform software fed switch active wdavc function wdavc_ft_show_all_flows_seg_ui`.

<#root>

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
					PROTO	PROTO	VLAN	SEC	NAME	TYPE	TYPE	SWAPPED
1	192.168.100.2	192.168.200.2	68	67	1	17	0	360	unknown	Full	Real Flow	Yes
SEG IDX	I/F ID	OPST I/F	SEG DIR	FIF DIR	Is SET	DOP ID	NFL HDL	BPS PND	APP PND	FRST TS	L	
0	9	----	Ingress	True	True	0	50331823	0	0	177403000	1	

Campos Llave

ID de interfaz: especifica el ID de interfaz


SEG DIR: Especifica la entrada de la dirección de salida

FIF DIR: Determina si ésta es o no la dirección del iniciador de flujo

NFL HDL: ID de flujo en hardware

Para ver la entrada en el hardware, ejecute el comando `show platform software fed switch active fnf flow-record ASIC`

num-flows

 Nota: Para elegir el ASIC, es la instancia ASIC a la que se asigna el puerto. Para identificar el ASIC, utilice los mapeos del comando `show platform software fed switch active|standby|member ifm`. El `start-index` se puede establecer en 0 si no está interesado en un flujo específico. De lo contrario, debe especificarse `start-index`. En el caso de los flujos de números, especifica el número máximo de flujos que se pueden visualizar: 10.

<#root>

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 = 0x00000000001216f, L2ByteCount = 0x000000001873006
```

Buscar varios errores y advertencias en la ruta de datos

Utilice el comando `show platform software fed switch active|standby|member wdv function wdv_ft_show_stats_ui | inc err|warn|fail` para ver posibles errores de la tabla de flujo:

<#root>

Switch#

```
show platform software fed switch active wdv function wdv_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
```

Utilice el comando `show platform software fed switch active wdvnc function wdvnc_stile_stats_show_ui | inc err` para ver cualquier error potencial de NBAR:

<#root>


Switch#

```
show platform software fed switch active wdvnc function wdvnc_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
```

Verificar que los Paquetes se Clonan en la CPU

Utilice el comando `show platform software fed switch active punt cpuq 21 | inc received` para verificar que los paquetes se clonan en la CPU para el procesamiento NBAR:

 Nota: En el laboratorio, este número no aumentó.

<#root>

Switch#

```
show platform software fed switch active punt cpuq 21 | inc received
```

Packets received from ASIC : 63

Identificar congestión de CPU

En tiempos de congestión, los paquetes se pueden descartar antes de enviarlos al proceso WDAVC. Utilice el comando `show platform software fed switch active wdavc function fed_wdavc_show_ots_stats_ui` para validar:

<#root>


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
```

 Sugerencia: Para borrar el contador de punt drop utilice el comando `show platform software fed switch active wdavc function fed_wdavc_clear_ots_stats_ui`.


Identificar problemas de escalabilidad

Si no hay entradas FNF libres en el hardware, el tráfico no está sujeto a la clasificación NBAR2.

Utilice el comando `show platform software fed switch active fnf sw-table-sizes asic`

shadow 0

para confirmar:

 Nota: Los flujos que se crean son específicos del switch y del núcleo básico cuando se crean. El número de switch (activo, en espera, etc.) debe especificarse en consecuencia. El número ASIC que se ingresa está vinculado a la interfaz respectiva, utilice `show platform software fed switch active|standby|member ifm mappings` para determinar el ASIC que corresponde a la interfaz. Para la opción de sombra, utilice siempre 0.

<#root>

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

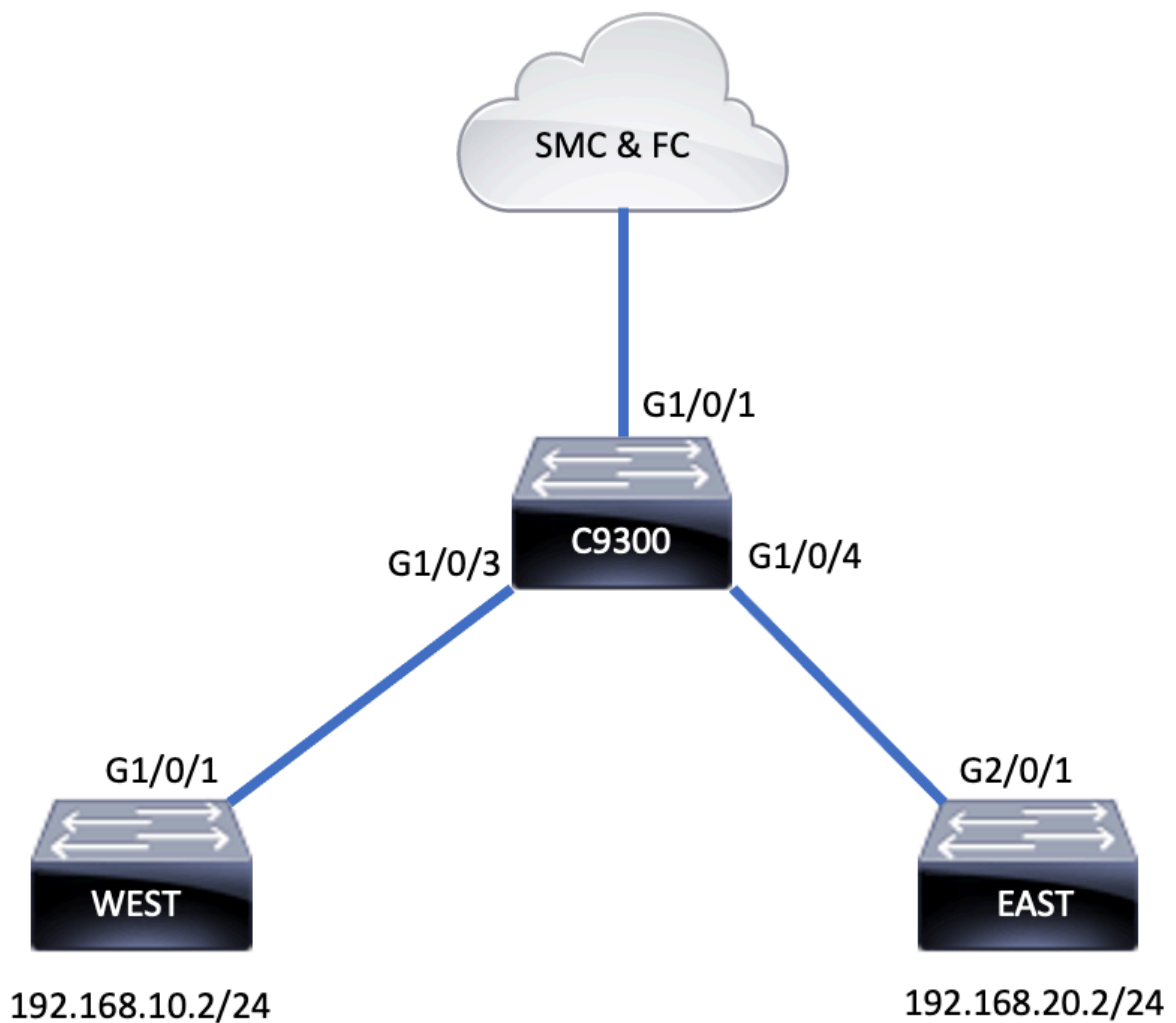
0	EGR	0	0	0	0
1	EGR	16640	0	0	1
2	EGR	0	0	0	0
3	EGR	16640	0	0	0
4	EGR	0	0	0	0
5	EGR	8192	0	0	1
6	EGR	0	0	0	0
7	EGR	0	0	0	0
8	EGR	0	0	0	0
9	EGR	0	0	0	0
10	EGR	0	0	0	0
11	EGR	0	0	0	0
12	EGR	0	0	0	0
13	EGR	0	0	0	0
14	EGR	0	0	0	0
15	EGR	0	0	0	0

Análisis de tráfico cifrado (ETA)

Antecedentes

- ETA se centra en la identificación de comunicaciones de malware en tráfico cifrado a través de la supervisión pasiva, la extracción de elementos de datos relevantes y una combinación de modelado de comportamiento y aprendizaje automatizado con seguridad global basada en la nube.
- ETA aprovecha la telemetría de NetFlow, así como la detección de malware cifrado y el cumplimiento criptográfico, y envía estos datos a Cisco StealthWatch.
- ETA extrae dos elementos de datos principales: el paquete de datos inicial (IDP) y la secuencia de duración y tiempo del paquete (SPLT).

Diagrama de la red




Componentes

ETA se compone de varios componentes diferentes que se utilizan conjuntamente para crear la solución ETA:

- NetFlow: Estándar que define los elementos de datos exportados por los dispositivos de red que describen los flujos en la red.
- Cisco StealthWatch: aprovecha el poder de la telemetría de red que incluye NetFlow, IPFIX, registros de proxy e inspección profunda de paquetes sin procesar para proporcionar visibilidad avanzada de la red, inteligencia de seguridad y análisis.
- Inteligencia cognitiva de Cisco: busca actividad maliciosa que ha eludido los controles de seguridad o a la que se ha accedido a través de canales no supervisados y dentro del entorno de una organización.
- Análisis de tráfico cifrado: función de Cisco IOS XE que utiliza algoritmos de comportamiento avanzados para identificar patrones de tráfico maliciosos a través del análisis de metadatos de flujo de entrada del tráfico cifrado, y detecta las amenazas potenciales ocultas en el tráfico cifrado.

 Nota: Esta parte del documento solo se centra en la configuración y verificación de ETA y

 NetFlow en el switch Catalyst serie 9000 y no cubre la implementación de StealthWatch Management Console (SMC) y Flow Collector (FC) en Cognitive Intelligence Cloud.

Restricciones

- El despliegue de ETA requiere DNA Advantage para funcionar
- ETA y un analizador de puerto conmutado (SPAN) de transmisión (TX) no son compatibles en la misma interfaz.

Esta no es una lista inclusiva, consulte la guía de configuración apropiada para el switch y la versión de código para todas las restricciones.

Configuración

Como se muestra en la salida, habilite ETA en el switch globalmente y defina el destino de exportación de flujo:

```
<#root>
C9300(config)#
et-analytics

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

 Sugerencia: DEBE utilizar el puerto 2055, no utilice otro número de puerto.

A continuación, configure Flexible NetFlow como se muestra en el resultado:

Configurar registro de flujo

```
<#root>
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
```

Configurar monitor de flujo

```
<#root>
```

```
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
```

Configurar registro de flujo

```
<#root>
```

```
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
```

Aplicar monitor de flujo

```
<#root>
```

```
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
```

Activar ETA en las interfaces del switch

```
<#root>
```

```
C9300(config)#
```

```
interface range g1/0/3-4
```

```
C9300(config-if-range)#
```

```
et-analytics enable
```

Verificación

Verifique que el monitor ETA, eta-mon, esté activo. Confirme que el estado esté asignado a través del comando `show flow monitor eta-mon`.

```
<#root>
```

```
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
```

Verifique que la memoria caché de ETA esté llena. Cuando NetFlow y ETA se configuran en la misma interfaz, utilice en `show flow monitor`

```
cache
```

lugar de `show flow monitor eta-mon cache` ya que la salida de `show flow monitor eta-mon cache` está vacía:

```
<#root>
```

```
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

Valide que los flujos se exportan hacia el SMC y el FC con el comando `show flow exporter eta-exp statistics`.

<#root>

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
```

Confirme que SPLT y IDP se exporten al FC con el comando `show platform software fed switch active fnf et-analytics-flows`.

```
<#root>
```

```
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
```

Valide qué interfaces se configuran para et-analytics con el comando `show platform software et-analytics interfaces`.

```
<#root>
```

```
C9300#
```

```
show platform software et-analytics interfaces
```

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

```
ET-Analytics VLANs
```

Utilice el comando `show platform software et-analytics global` para ver un estado global de ETA:

```
<#root>
```

```
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
```

Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).