

# 驗證Catalyst 9000交換器上的安全ACL

## 目錄

---

### [簡介](#)

#### [必要條件](#)

[需求](#)

[採用元件](#)

#### [背景資訊](#)

[技術](#)

#### [ACL資源利用率示例](#)

[範例 1.IPv4 TCAM](#)

[範例 2.IPv4 TCAM/L4OP/VCU](#)

[範例 3.IPv6TCAM/L4OP/VCU](#)

### [拓撲](#)

#### [設定和驗證](#)

[案例 1.PACL\(IP ACL\)](#)

[使用IP ACL配置PACL](#)

[驗證PACL](#)

[案例 2.PACL\(MAC ACL\)](#)

[使用MAC ACL配置PACL](#)

[驗證PACL](#)

[案例 3.RACL](#)

[配置RACL](#)

[驗證RACL](#)

[案例 4.VACL](#)

[配置VACL](#)

[驗證VACL](#)

[案例 5.群組/使用者端ACL\(DACL\)](#)

[配置GACL](#)

[檢驗GACL](#)

[案例 6.ACL記錄](#)

#### [疑難排解](#)

[ACL統計資訊](#)

[清除ACL統計資訊](#)

[ACL TCAM用完後會發生什麼情況？](#)

[ACL TCAM耗盡](#)

[VCU耗盡](#)

[ACL系統日誌錯誤](#)

[資源外情形和恢復操作](#)

[檢驗ACL規模](#)

[自定義SDM模板 \( TCAM重新分配 \)](#)

#### [相關資訊](#)

[Debug和Trace命令](#)

---

# 簡介

本文說明如何驗證Catalyst 9000系列交換器上的ACL (存取控制清單) 及對其進行疑難排解。

## 必要條件

### 需求

本文件沒有特定需求。

### 採用元件

本檔案中的資訊是根據以下硬體版本：

- C9200
- C9300
- C9400
- C9500
- C9600

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。



注意:有關在其他思科平台上啟用這些功能的命令，請參閱相應的配置指南。

## 背景資訊

ACL會在流量通過路由器或交換機時過濾流量，並允許或拒絕通過指定介面的資料包。ACL是適用於封包的允許和拒絕條件的順序集合。當封包在介面上收到時，交換器會根據存取清單中指定的標準，將封包中的欄位與任何套用的ACL進行比較，以驗證封包是否具有所需的轉送許可權。它會根據訪問清單中的條件逐一測試資料包。第一個符專案會決定交換器是接受還是拒絕封包。因為交換器在第一個相符專案出現後停止測試，所以條件在清單中的順序非常重要。如果沒有相符的條件，交換器就會拒絕封包。如果沒有限制，交換器會轉送封包；否則交換器會捨棄封包。交換器可以對其轉送的所有封包使用ACL。

您可以設定存取清單，以便為網路提供基本安全性。如果不配置ACL，則允許通過交換機的所有資料包到達所有網路部分。您可以使用ACL控制哪些主機可以訪問網路的不同部分，或者決定哪些型別的流量會在路由器介面上轉發或阻止。例如，您可以轉發電子郵件流量，但不能轉發Telnet流量。

### 技術

ACE	存取控制專案(ACE)- ACL中的單一規則/線路
-----	---------------------------

ACL	訪問控制清單(ACL) — 一組應用於埠的ACE
DAACL	可下載ACL(DAACL) — 通過ISE安全策略動態推送的ACL
PAACL	連線埠ACL(PAACL) — 應用於第2層介面的ACL
RACL	路由ACL(RACL) — 應用於第3層介面的ACL
VACL	VLAN ACL(VACL) — 應用於VLAN的ACL
GACL	組ACL(GACL) — 根據使用者組或客戶端的身份動態分配的ACL
IP ACL	用於對IPv4/IPv6資料包進行分類。這些規則包含各種第3層和第4層資料包欄位和屬性，包括但不限於源和目標IPv4地址、TCP/UDP源和目標埠、TCP標誌和DSCP等。
MACL	Mac Address ACL(MACL) — 用於對非IP資料包進行分類。規則包含各種第2層欄位和屬性，包括源/目標MAC地址、乙太網型別等。
L4OP	第4層運算子埠(L4OP) — 匹配除EQ (等於) 之外的邏輯。GT (大於)、LT (小於)、NE (不等於) 和RANGE (從至)
VCU	值比較單元(VCU)- L4OP轉換為VCU，以便對第4層報頭執行分類
VMR	值掩碼結果(VMR)- ACE條目在TCAM中作為VMR進行內部程式設計。
CGD	類別組資料庫(CGD)- FMAN-FP在其中儲存ACL內容
類	如何在CGD中識別ACE
CG	Class Group(CG) — 一組有關如何在CGD中標識ACL的類
CGE	類組條目(CGE) — 儲存在類組中的ACE條目
FMAN	轉發管理器(FMAN)- Cisco IOS® XE與硬體之間的程式設計層
FED	轉發引擎驅動程式(FED) — 對裝置硬體進行程式設計的元件

# ACL資源利用率示例

此處有三個範例來說明ACL如何使用TCAM、L4OP和VCU。

## 範例 1.IPv4 TCAM

```
access-list 101 permit ip any 10.1.1.0 0.0.0.255
access-list 101 permit ip any 10.1.2.0 0.0.0.255
access-list 101 permit ip any 10.1.3.0 0.0.0.255
access-list 101 permit ip any 10.1.4.0 0.0.0.255
access-list 101 permit ip any 10.1.5.0 0.0.0.255
```

	TCAM條目	L4OP	VCU
消費	5	0	0

## 範例 2.IPv4 TCAM/L4OP/VCU

```
ip access-list extended TEST
permit tcp 192.168.1.0 0.0.0.255 any ne 3456
permit tcp 10.0.0.0 0.255.255.255 any range 3000 3100
permit tcp 172.16.0.0 0.0.255.255 any range 4000 8000
permit tcp 192.168.2.0 0.0.0.255 gt 10000 any eq 20000
```

Each range L4OPs consume two VCU

Source and destination L4OPs consume separate VCUs

<#root>

```
ip access-list extended TEST
10 permit tcp 192.168.1.0 0.0.0.255 any
neq 3456
```

<-- 1 L4OP, 1 VCU

```
20 permit tcp 10.0.0.0 0.255.255.255 any
```

```

range 3000 3100 <-- 1 L4OP, 2 VCU

30 permit tcp 172.16.0.0 0.0.255.255 any

range 4000 8000 <-- 1 L4OP, 2 VCU

40 permit tcp 192.168.2.0 0.0.0.255

gt 10000

any

eq 20000 <-- 2 L4OP, 2 VCU

```

	TCAM條目	L4OP	VCU
消費	4	5	7

### 範例 3.IPv6 TCAM/L4OP/VCU

IPv6 ACE使用兩個TCAM條目，而不是一個IPv4條目。在本例中，四個ACE消耗八個TCAM，而不是四個。

```
<#root>
```

```

ipv6 access-list v6TEST
sequence 10 deny ipv6 any 2001:DB8:C18::/48 fragments
sequence 20 deny ipv6 2001:DB8::/32 any
sequence 30 permit tcp host 2001:DB8:C19:2:1::F host 2001:DB8:C18:2:1::1

eq bgp <-- One L4OP & VCU

sequence 40 permit tcp host 2001:DB8:C19:2:1::F

eq bgp

host 2001:DB8:C18:2:1::1

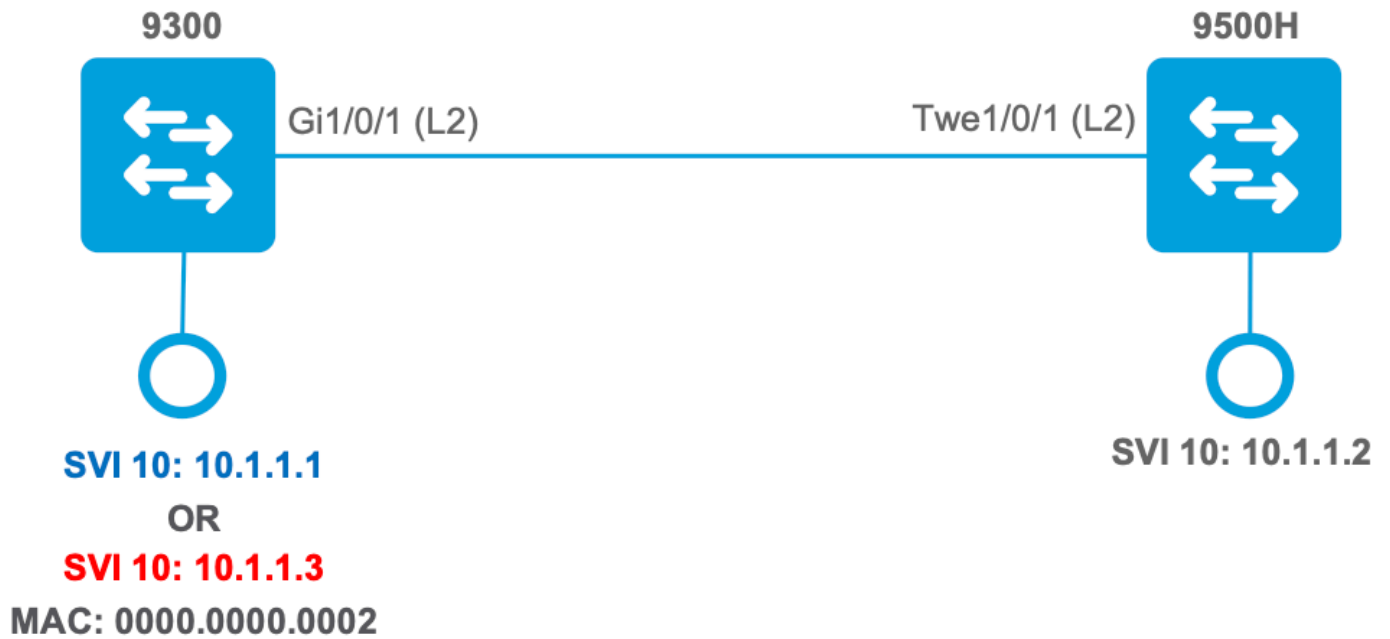
<-- One L4OP & VCU

```

	TCAM條目	L4OP	VCU
消費	8	2	2

## 拓撲

9300 VLAN 10 SVI根據示例中顯示了轉發或丟棄結果，使用本圖中所示的兩個IP地址之一。



## 設定和驗證

本節介紹如何驗證軟體和硬體中的ACL程式設計並對其進行故障排除。

### 案例 1.PACL(IP ACL)

PACL被分配到第2層介面。

- 安全邊界：埠或VLAN
- 附件：第2層介面
- 方向：入口或出口（一次一個）
- 支援的ACL型別：MAC ACL和IP ACL（標準或擴展）

使用IP ACL配置PACL

```
<#root>
9500H(config)#
ip access-list extended TEST          <-- Create a named extended ACL

9500H(config-ext-nacl)#
permit ip host 10.1.1.1 any

9500H(config-ext-nacl)#
permit udp host 10.1.1.1 eq 1000 host 10.1.1.2
```

9500H#

show access-lists TEST <-- Display the ACL configured

```
Extended IP access list TEST
 10 permit ip host 10.1.1.1 any
 20 permit udp host 10.1.1.1 eq 1000 host 10.1.1.2
```

9500H(config)#

interface twentyFiveGigE 1/0/1 <-- Apply ACL to Layer 2 interface

9500H(config-if)#

ip access-group TEST in

9500H#

show running-config interface twentyFiveGigE 1/0/1

Building configuration...

Current configuration : 63 bytes

```
!
interface TwentyFiveGigE1/0/1
```

```
 ip access-group TEST in <-- Display the ACL applied to the interface
```

end

## 驗證PACL

檢索與介面關聯的IF\_ID。

<#root>

9500H#

show platform software fed active ifm interfaces ethernet

Interface

IF\_ID

State

-----  
TwentyFiveGigE1/0/1

0x00000008

READY

```
<-- IF_ID value for Tw1/0/1
```

驗證繫結到IF\_ID的類組ID(CG ID)。

```
<#root>
```

```
9500H#
```

```
show platform software fed active acl interface 0x8 <-- IF_ID with leading zeros omitted
```

```
#####
#####
##### Printing Interface Infos #####
#####
#####
```

```
INTERFACE:
```

```
TwentyFiveGigE1/0/1 <-- Confirms the interface matches the IF_ID
```

```
MAC 0000.0000.0000
```

```
#####
intfinfo: 0x7f8cfc02de98
Interface handle: 0x7e000028
```

```
Interface Type: Port <-- Type: Port indicates Layer 2 interface
```

```
if-id: 0x0000000000000008 <-- IF_ID 0x8 is correct
```

```
Input IPv4: Policy Handle: 0x5b000093
```

```
Policy Name: TEST <-- The named ACL bound to this interface
```

```
CG ID: 9 <-- Class Group ID for this entry
```

```
CGM Feature: [0] acl <-- Feature is ACL
```

```
Bind Order: 0
```

與CG ID關聯的ACL資訊。

```
<#root>
```



9500H#

show platform software fed active acl info acl-cgid 9 <-- The CG ID associated to the ACL TEST

```
#####  
#####  
##### Printing CG Entries #####  
#####  
#####  
#####  
=====
```

ACL CG (acl/9): TEST type: IPv4 <-- feature ACL/CG ID 9: ACL name TEST : ACL type IPv4

Total Ref count 1

-----  
1 Interface

<-- ACL is applied to one interface

-----  
region reg\_id: 10  
subregion subr\_id: 0  
GCE#:1

#flds: 2

14:N

matchall:N deny:N

<-- #flds: 2 = two fields in entry | 14:N (no Layer 4 port match)

Result: 0x01010000

ipv4\_src: value

=

0x0a010101

,

mask = 0xffffffff

<-- src 0x0a010101 hex = 10.1.1.1 | mask 0xffffffff = exact host match

ipv4\_dst: value

=

0x00000000, mask = 0x00000000

```

<--

dst & mask = 0x00000000 = match any
    GCE#:1 #flds: 4
14:Y
    matchall:N deny:N
<-- #flds: 4 = four fields in entry | 14:Y (ACE uses UDP port L4 match)

    Result: 0x01010000

ipv4_src: value = 0x0a010101, mask = 0xffffffff <-- Exact match (host) 10.1.1.1

ipv4_dst: value = 0x0a010102, mask = 0xffffffff <-- Exact match (host) 10.1.1.2

ip_prot: start = 17, end = 17 <-- protocol 17 is UDP

l4_src: start = 1000, end = 1000 <-- matches eq 1000 (equal UDP port 1000)

```

有關CG ID以及哪些介面使用CG ID的策略資訊。

```

<#root>
9500H#
show platform software fed active acl policy 9 <-- Use the CG ID value

#####
#####          #####
##### Printing Policy Infos #####
#####          #####
#####          #####
#####          #####

INTERFACE: TwentyFiveGigE1/0/1 <-- Interface with ACL applied

MAC 0000.0000.0000
#####
    intfinfo: 0x7f8cfc02de98
    Interface handle: 0x7e000028
    Interface Type: Port

if-id: 0x0000000000000008 <-- The Interface IF_ID 0x8

```

-----

Direction: Input

<-- ACL is applied in the ingress direction

Protocol Type:IPv4

<-- Type is IPv4

Policy Intface Handle: 0x880000c1  
Policy Handle: 0x5b000093

#####  
#####  
##### Policy information #####  
#####  
#####

Policy handle : 0x5b000093

Policy name : TEST

<-- ACL Name TEST

ID : 9

<-- CG ID for this ACL entry

Protocol : [3] IPV4

Feature : [1] AAL\_FEATURE\_PACL

<-- ASIC feature is PAACL

Number of ACLs : 1

#####  
## Complete policy ACL information  
#####  
Acl number : 1

=====  
Acl handle : 0x320000d2  
Acl flags : 0x00000001

Number of ACES

: 3

<-- 3 ACES: two explicit and the implicit deny entry

Ace handle [1] : 0xb700010a

Ace handle [2] : 0x5800010b

Interface(s):

TwentyFiveGigE1/0/1


<-- The interface ACL is applied

#####  
#####  
##### Policy instance information #####  
#####  
#####

Policy intf handle : 0x880000c1  
Policy handle : 0x5b000093

```
ID : 9
Protocol : [3] IPV4
Feature : [1] AAL_FEATURE_PACL
Direction : [1] Ingress
Number of ACLs : 1
Number of VMRs : 3-----
```

確認PACL工作正常。

 附註：當您輸入 `show ip access-lists privileged EXEC` 命令時，顯示的匹配計數不會計算硬體中訪問控制的資料包。使用 `show platform software feed switch {switch_num|active|standby}acl counters hardware privileged EXEC` 命令可獲取交換和路由封包的一些基本硬體ACL統計資料。

<#root>

```
### Ping originated from neighbor device with source 10.1.1.1 ###
```

```
C9300#
```

```
ping 10.1.1.2 source g 1/0/1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
```

```
Packet sent with a source address of 10.1.1.1
```

```
<--- Ping source is permitted and p
```

```
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms <-- 100% ping success
```

```
### Ping originated from neighbor device with source 10.1.1.3 ###
```

```
C9300#
```

```
ping 10.1.1.2 source g 1/0/1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
```

```
Packet sent with a source address of 10.1.1.3
```

```
<-- Ping source is denied (implicit
```

```
.....
```

```
Success rate is 0 percent (0/5)
```

```
<-- 0% ping success
```

```
### Confirm PACL drop ###
```

```
9500H#
```

```
show access-lists TEST
```

Extended IP access list TEST

```
10 permit ip host 10.1.1.1 any <-- Counters in this command do not
20 permit udp host 10.1.1.1 eq 1000 host 10.1.1.2
```

9500H#

```
show platform software fed active acl counters hardware | i PAcl Drop
Ingress IPv4 PAcl Drop (0x77000005): 11 frames <-- Hardware level command displays
Ingress IPv6 PAcl Drop (0x12000012): 0 frames
```

<...snip...>

## 案例 2.PACL(MAC ACL)

PACL被分配到第2層介面。

- 安全邊界：埠或VLAN
- 附件：第2層介面
- 方向：入口或出口 (一次一個)
- 支援的ACL型別：MAC ACL和IP ACL (標準或擴展)

使用MAC ACL配置PAcl

<#root>

9500H#

```
show run | sec mac access-list
```

```
mac access-list extended
```

```
MAC-TEST <-- MAC ACL named MAC-TEST
```

```
permit host 0001.aaaa.aaaa any <-- permit host MAC to any dest MAC
```

9500H#

```
show access-lists MAC-TEST
```

```
Extended MAC access list MAC-TEST
  permit host 0001.aaaa.aaaa any
```

9500H#

```
show running-config interface twentyFiveGigE 1/0/1
```

Building configuration...

```
interface TwentyFiveGigE1/0/1
switchport access vlan 10
switchport mode access
```

```
mac access-group MAC-TEST in <-- Applied MACL to layer 2 interface
```

### 驗證PACL

檢索與介面關聯的IF\_ID。

<#root>

9500H#

```
show platform software fed active ifm interfaces ethernet
```

Interface

IF\_ID

State

-----  
TwentyFiveGigE1/0/1

0x00000008

READY

<-- IF\_ID value for Tw1/0/1

驗證繫結到IF\_ID的類組ID(CG ID)。

<#root>

9500H#

```
show platform software fed active acl interface 0x8 <-- IF_ID with leading zeros omitted
```

```
#####
#####
##### Printing Interface Infos #####
#####
#####
```

INTERFACE: TwentyFiveGigE1/0/1 <-- Confirms the interface matches the IF

MAC 0000.0000.0000

#####

intfinfo: 0x7f489404e408  
Interface handle: 0x7e000028

Interface Type: Port <-- Type: Port indicates Layer 2 interface

if-id: 0x0000000000000008 <-- IF\_ID 0x8 is correct

Input MAC: Policy Handle: 0xde000098

Policy Name: MAC-TEST <-- The named ACL bound to this interface

CG ID: 20 <-- Class Group ID for this entry

CGM Feature: [0] acl <-- Feature is ACL

Bind Order: 0

與CG ID關聯的ACL資訊。

<#root>

9500H#

show platform software fed active acl info acl-cgid 20 <-- The CG ID associated to the ACL MAC-TEST

#####  
#####  
##### Printing CG Entries #####  
#####  
#####  
#####  
=====

ACL CG (acl/20): MAC-TEST type: MAC <-- feature ACL/CG ID 20: ACL name MAC-TEST

Total Ref count 1

1 Interface <-- Applied to one interface

-----  
region reg\_id: 3  
subregion subr\_id: 0  
GCE#:1 #flds: 2 l4:N matchall:N deny:N  
Result: 0x01010000

```

mac_dest: value = 0x00, mask = 0x00          <-- Mac dest: hex 0x00 mask 0x00 is "any destination"

mac_src: value = 0x1aaaaaaaaa

,

mask = 0xffffffffffff

<-- Mac source: 0x1aaaaaaaaa | hex with leading zeros omitted (0001.aaaa.aaaa) & mask 0xffffffffffff is 1

```

有關CG ID以及哪些介面使用CG ID的策略資訊。

```
<#root>
```

```
9500H#
```

```
show platform software fed active acl policy 20          <-- Use the CG ID value
```

```

#####
#####          #####
#####      Printing Policy Infos      #####
#####          #####
#####          #####
#####

```

```
INTERFACE: TwentyFiveGigE1/0/1          <-- Interface with ACL applied
```

```

MAC 0000.0000.0000
#####
  intfinfo: 0x7f8cfc02de98
  Interface handle: 0x7e000028
  Interface Type: Port

```

```
if-id: 0x0000000000000008          <-- The Interface IF_ID 0x8
```

```

-----
Direction:  Input          <-- ACL is applied in the ingress direction

```

```
Protocol Type:MAC          <-- Type is MAC
```

```

  Policy Intface Handle: 0x30000c6
  Policy Handle: 0xde000098

```

```

#####
#####          #####
#####      Policy information      #####
#####          #####
#####
Policy handle          : 0xde000098

```



```

Policy name          : MAC-TEST                               <-- ACL name is MAC-TEST

ID                  : 20                                       <-- CG ID for this ACL entry

Protocol            : [1] MAC

Feature             : [1] AAL_FEATURE_PACL                     <-- ASIC Feature is PACL

Number of ACLs      : 1

#####
## Complete policy ACL information
#####
Acl number : 1
=====
Acl handle : 0xd60000dc
Acl flags  : 0x00000001

Number of ACEs : 2                                           <-- 2 ACEs: one permit, and one implicit deny

    Ace handle [1] : 0x38000120
    Ace handle [2] : 0x31000121

Interface(s):

    TwentyFiveGigE1/0/1                                       <-- Interface the ACL is applied

#####
#####
##### Policy instance information #####
#####
#####
Policy intf handle   : 0x030000c6
Policy handle        : 0xde000098
ID                   : 20
Protocol             : [1] MAC
Feature              : [1] AAL_FEATURE_PACL
Direction            : [1] Ingress
Number of ACLs       : 1
Number of VMRS       : 3-----

```

確認PACL工作正常：

- MACL僅允許源地址0001.aaaa.aaaa。
- 由於這是MAC ACL，因此會丟棄非IP ARP資料包，從而導致ping失敗。

<#root>

### Ping originated from neighbor device with Source MAC 0000.0000.0002 ###

C9300#

ping 10.1.1.2 source vlan 10

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

Packet sent with a source address of 10.1.1.1

.....

Success rate is 0 percent (0/5)

C9300#

show ip arp

Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	10.1.1.2	0			

Incomplete

ARPA

<-- ARP is unable to complete on Source device

### Monitor capture configured on Tw 1/0/1 ingress ###

9500H#

monitor capture 1 interface TwentyFiveGigE 1/0/1 in match any

9500H#

show monitor cap

Status Information for Capture 1

Target Type:

Interface: TwentyFiveGigE1/0/1, Direction: IN

9500H#sh monitor capture 1 buffer brief | inc ARP

5 4.767385 00:00:00:00:00:02 b^FAR

ff:ff:ff:ff:ff:ff ARP 60 Who has 10.1.1.2? Tell 10.1.1.1

8 8.767085 00:00:00:00:00:02 b^FAR ff:ff:ff:ff:ff:ff ARP 60 Who has 10.1.1.2? Tell 10.1.1.1

11 10.767452 00:00:00:00:00:02 b^FAR ff:ff:ff:ff:ff:ff ARP 60 Who has 10.1.1.2? Tell 10.1.1.1

13 12.768125 00:00:00:00:00:02 b^FAR ff:ff:ff:ff:ff:ff ARP 60 Who has 10.1.1.2? Tell 10.1.1.1

<-- 9300 (10.1.1.1) sends ARP request, but since there is no reply 4 more ARP requests are sent

9500H#

show platform software fed active acl counters hardware | inc MAC PAcl Drop

Ingress MAC PAcl Drop (0x73000021): 937 frames

<-- Confirmed that ARP request

Egress MAC PAcl Drop (0x0200004c): 0 frames

<...snip...>

## 案例 3.RACL

RACL分配給第3層介面，例如SVI或路由介面。

- 安全邊界：不同的子網
- 附件：第3層介面
- 方向：Ingress或Egress
- 支援的ACL型別：IP ACL ( 標準或擴展 )

### 配置RACL

<#root>

```
9500H(config)#
ip access-list extended TEST          <-- Create a named extended ACL

9500H(config-ext-nacl)#
permit ip host 10.1.1.1 any

9500H(config-ext-nacl)#
permit udp host 10.1.1.1 eq 1000 host 10.1.1.2

9500H#
show access-lists TEST                <-- Display the ACL configured

Extended IP access list TEST
 10 permit ip host 10.1.1.1 any
 20 permit udp host 10.1.1.1 eq 1000 host 10.1.1.2

9500H(config)#
interface Vlan 10                      <-- Apply ACL to Layer 3 SVI interface

9500H(config-if)#
ip access-group TEST in

9500H#
show running-config interface Vlan 10

Building configuration...

Current configuration : 84 bytes
!
```

```

interface Vlan10
  ip access-group TEST in <-- Display the ACL applied to the interface
end

```

驗證RACL

檢索與介面關聯的IF\_ID。

```

<#root>
9500H#
show platform software fed active ifm mappings l3if-le <-- Retrieve the IF_ID for a Layer 3 SVI type po
Mappings Table
L3IF_LE          Interface          IF_ID          Type
-----
0x00007f8d04983958
Vlan10
0x00000026
      SVI_L3_LE
<-- IF_ID value for SVI 10

```

驗證繫結到IF\_ID的類組ID(CG ID)。

```

<#root>
9500H#
show platform software fed active acl interface 0x26 <-- IF_ID for SVI Vlan 10 with leading zeros omitted

#####
#####          #####
#####   Printing Interface Infos   #####
#####          #####
#####

INTERFACE: Vlan10 <-- Confirms the interface matches the IF_ID

MAC 0000.0000.0000
#####
  intfinfo: 0x7f8cfc02de98
  Interface handle: 0x6e000047

```

Interface Type: L3 <-- Type: L3 indicates Layer 3 type interface

if-id: 0x0000000000000026 <-- IF\_ID 0x26 is correct

Input IPv4: Policy Handle: 0x2e000095

Policy Name: TEST <-- The named ACL bound to this interface

CG ID: 9 <-- Class Group ID for this entry

CGM Feature: [0] acl <-- Feature is ACL

Bind Order: 0

與CG ID關聯的ACL資訊。

<#root>

9500H#

show platform software fed active acl info acl-cgid 9 <-- The CG ID associated to the ACL TEST

#####
#####
##### Printing CG Entries #####
#####
#####
=====

ACL CG (acl/9): TEST type: IPv4

<-- feature ACL/CG ID 9: ACL name TEST : ACL type IPv4

Total Ref count 2

2 Interface

<-- Interface count is 2. Applied to SVI 10 and as PACL to Tw1/0

region reg\_id: 10
subregion subr\_id: 0
GCE#:1

#flds: 2

14:N

matchall:N deny:N

<-- #flds: 2 = two fields in entry | 14:N (no Layer 4 port match)

Result: 0x01010000

ipv4\_src: value

=

0x0a010101

,

mask = 0xffffffff

<-- src 0x0a010101 hex = 10.1.1.1 | mask 0xffffffff = exact host match

ipv4\_dst: value

=

0x00000000, mask = 0x00000000

<--

dst & mask = 0x00000000 = match any

GCE#:1 #flds: 4

14:Y

matchall:N deny:N

<-- #flds: 4 = four fields in entry | 14:Y (ACE uses UDP port L4 match)

Result: 0x01010000

ipv4\_src: value = 0x0a010101, mask = 0xffffffff <-- Exact match (host) 10.1.1.1

ipv4\_dst: value = 0x0a010102, mask = 0xffffffff <-- Exact match (host) 10.1.1.2

ip\_prot: start = 17, end = 17

<-- protocol 17 is UDP

l4\_src: start = 1000, end = 1000

<-- matches eq 1000 (equal UDP port 1000)

有關CG ID以及哪些介面使用CG ID的策略資訊。

<#root>

9500H#

show platform software fed active acl policy 9 <-- Use the CG ID Value

```
#####  
#####  
##### Printing Policy Infos #####  
#####  
#####
```

INTERFACE: Vlan10 <-- Interface with ACL applied

MAC 0000.0000.0000

```
#####  
intfinfo: 0x7f8cfc02de98  
Interface handle: 0x6e000047  
Interface Type: L3
```

if-id: 0x0000000000000026 <-- Interface IF\_ID 0x26

-----

Direction: Input <-- ACL applied in the ingress direction

Protocol Type:IPv4 <-- Type is IPv4

Policy Intfance Handle: 0x1c0000c2  
Policy Handle: 0x2e000095

```
#####  
#####  
##### Policy information #####  
#####  
#####
```

Policy handle : 0x2e000095

Policy name : TEST <-- ACL name TEST

ID : 9

<-- CG ID for this ACL entry

Protocol : [3] IPV4

Feature : [27] AAL\_FEATURE\_RACL <-- ASIC feature is RACL

Number of ACLs : 1

```
#####
## Complete policy ACL information
#####
Acl number      : 1
=====
Acl handle      : 0x7c0000d4
Acl flags       : 0x00000001

Number of ACEs   : 5                                <-- 5 Aces: 2 explicit, 1 implicit deny, 2 ???

Ace handle [1]  : 0x0600010f
Ace handle [2]  : 0x8e000110
Ace handle [3]  : 0x3b000111
Ace handle [4]  : 0xeb000112
Ace handle [5]  : 0x79000113
```

Interface(s):


Vlan10

<-- The interface the ACL is applied

```
#####
##### Policy instance information #####
#####
#####
Policy intf handle : 0x1c0000c2
Policy handle      : 0x2e000095
ID                 : 9
Protocol           : [3] IPV4
Feature            : [27] AAL_FEATURE_RACL
Direction          : [1] Ingress
Number of ACLs     : 1
Number of VMRs     : 4-----
```

確認RACL工作正常。

---

 附註：當您輸入 show ip access-lists privileged EXEC 命令時，顯示的匹配計數不會計算硬體中訪問控制的資料包。使用 show platform software fed switch{switch\_num|active|standby}acl 計數器硬體取得交換和路由封包的一些基本硬體ACL統計資料。

---

<#root>

```
### Ping originated from neighbor device with source 10.1.1.1 ###
```

C9300#

```
ping 10.1.1.2 source g 1/0/1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

```
Packet sent with a source address of 10.1.1.1
```

<--- Ping source is permitted and p



!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms <-- 100% ping success

### Ping originated from neighbor device with source 10.1.1.3 ###

C9300#

ping 10.1.1.2 source g 1/0/1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:

Packet sent with a source address of 10.1.1.3

<-- Ping source is denied (implicit deny)

.....

Success rate is 0 percent (0/5)

<-- 0% ping success

### Confirm RACL drop ###

9500H#

show access-lists TEST

Extended IP access list TEST

10 permit ip host 10.1.1.1 any

<-- Counters in this command do not apply

20 permit udp host 10.1.1.1 eq 1000 host 10.1.1.2

9500H#

show platform software fed active acl counters hardware | i RACL Drop

Ingress IPv4 RACL Drop

(0xed000007):

100 frames <-- Hardware level command display

<...snip...>

## 案例 4.VACL

VACL被分配到第2層VLAN。

- 安全邊界：在VLAN內或跨VLAN
- 附件：VLAN/VLAN對映
- Direction：一次輸入和輸出
- 支援的ACL型別：MAC ACL和IP ACL（標準或擴展）

配置VACL

<#root>

```
ip access-list extended TEST

10 permit ip host 10.1.1.1 any
20 permit ip any host 10.1.1.1
```

```
ip access-list extended ELSE
```

```
10 permit ip any any
```

```
vlan access-map VACL 10
```

```
match ip address TEST
action forward
```

```
vlan access-map VACL 20
```

```
match ip address ELSE
action drop
```

```
vlan filter VACL vlan-list 10
```

```
9500H#
```

```
sh vlan access-map VACL
```

```
Vlan access-map "VACL" 10
```

```
Match clauses:
  ip address: TEST
```

```
Action:
```

```
forward
```

```
Vlan access-map "VACL" 20
```

```
Match clauses:
  ip address: ELSE
```

```
Action:
```

```
drop
```

```
9500H#
```

```
sh vlan filter access-map VACL
```

```
VLAN Map VACL is filtering VLANs:
```

## 驗證VACL

檢索與介面關聯的IF\_ID。

```
<#root>
```

```
9500H#
```

```
show platform software fed active ifm interfaces vlan
```

```
Interface
```

```
IF_ID
```

```
State
```

```
-----  
Vlan10                                0x00420010
```

```
READY
```

驗證繫結到IF\_ID的類組ID(CG ID)。

```
<#root>
```

```
9500H#
```

```
show platform software fed active acl interface 0x420010 <-- IF_ID for the Vlan
```

```
#####  
#####  
##### Printing Interface Infos #####  
#####  
#####  
#####
```

```
INTERFACE: Vlan10
```

```
<-- Can be L2 only, with no vlan interfa
```

```
MAC 0000.0000.0000
```

```
#####  
intfinfo: 0x7fc8cc7c7f48  
Interface handle: 0xf1000024  
Interface Type: Vlan  
if-id: 0x0000000000420010
```

```
Input IPv4:
```

```
Policy Handle: 0xd10000a3
```

```
<-- VACL has both Ingress and Egress actions
```

```
Policy Name: VACL
```

```
<-- Name of the VACL used
```

CG ID: 530

<-- Class Group ID for entry

CGM Feature: [35] acl-grp

<-- Feature is ACL group, versus ACL

Bind Order: 0

Output IPv4:

Policy Handle: 0xc8000a4

<-- VACL has both Ingress and Egress actions

Policy Name: VACL

CG ID: 530

CGM Feature: [35] acl-grp

Bind Order: 0

與CG組ID關聯的ACL資訊。

同一個命名VACL策略中使用了兩個ACL，它們分組到此acl組中

<#root>

9500H#

show platform software fed active acl info acl-grp-cgid 530 <-- use the group-id command versus gc ID

```
#####
#####
##### Printing CG Entries #####
#####
#####
#####
```

=====

ACL CG (acl-grp/530): VACL type: IPv4

<-- feature acl/group ID 530: name VA

Total Ref count 2

-----

2 VACL

<-- Ingress and egress ACL direction

-----

region reg\_id: 12

subregion subr\_id: 0

GCE#:10 #flds: 2 14:N matchall:N deny:N

Result: 0x06000000

ipv4\_src: value = 0x0a010101, mask = 0xffffffff

<-- permit from host 10.1.1.1 (see PACL exampl

```

ipv4_dst: value = 0x00000000, mask = 0x00000000          <-- to any other host

      GCE#:20 #flds: 2 14:N matchall:N deny:N
      Result: 0x06000000

ipv4_src: value = 0x00000000, mask = 0x00000000          <-- permit from any host

ipv4_dst: value = 0x0a010101, mask = 0xffffffff          <-- to host 10.1.1.1

      GCE#:10 #flds: 2 14:N matchall:N deny:N
      Result: 0x05000000

ipv4_src: value = 0x00000000, mask = 0x00000000          <-- This is the ACL named 'ELSE' which is per

      ipv4_dst: value = 0x00000000, mask = 0x00000000          <-- with VACL, the logic used was "per

```

有關CG ID以及哪些介面使用CG ID的策略資訊。

```

<#root>
9500H#
show platform software fed active acl policy 530          <-- use the acl-grp ID

#####
#####          #####
#####    Printing Policy Infos          #####
#####          #####
#####          #####

INTERFACE: Vlan10
MAC 0000.0000.0000
#####
      intfinfo: 0x7fa15802a5d8
      Interface handle: 0xf1000024

Interface Type: Vlan          <-- Interface type is the Vlan, not a specific id

if-id: 0x0000000000420010          <-- the Vlan IF_ID matches Vlan 10

-----

Direction: Input          <-- VACL in the input direction

```

```

Protocol Type:IPv4
  Policy Interface Handle: 0x44000001
  Policy Handle: 0x29000090

#####
#####
##### Policy information #####
#####
#####
Policy handle      : 0x29000090

Policy name       : VACL                                <-- the VACL policy is named 'VACL'

ID                : 530
Protocol          : [3] IPV4

Feature           : [23] AAL_FEATURE_VACL             <-- ASIC feature is VACL

Number of ACLs    : 2                                <-- 2 ACL used in the VACL: "TEST & ELSE"

#####
## Complete policy ACL information
#####
Acl number : 1
=====
Acl handle : 0xa6000090
Acl flags  : 0x00000001
Number of ACEs : 4
  Ace handle [1] : 0x87000107
  Ace handle [2] : 0x30000108
  Ace handle [3] : 0x73000109
  Ace handle [4] : 0xb700010a

Acl number : 2
=====
Acl handle : 0x0f000091
Acl flags  : 0x00000001
Number of ACEs : 1
  Ace handle [1] : 0x5800010b

Interface(s):
  Vlan10
#####
##### Policy instance information #####
#####
#####
Policy intf handle : 0x44000001
Policy handle      : 0x29000090

ID                : 530                                <-- 530 is the acl group ID

Protocol          : [3] IPV4
Feature           : [23] AAL_FEATURE_VACL

Direction        : [1] Ingress                        <-- Ingress VACL direction

Number of ACLs    : 2

```

Number of VMRs : 4-----  
Direction: Output  
Protocol Type:IPv4  
Policy Interface Handle: 0xac000002  
Policy Handle: 0x31000091

```
#####  
#####  
##### Policy information #####  
#####  
#####  
Policy handle : 0x31000091  
Policy name : VACL  
ID : 530  
Protocol : [3] IPV4  
Feature : [23] AAL_FEATURE_VACL  
Number of ACLs : 2
```

```
#####  
## Complete policy ACL information  
#####  
ACL number : 1
```

```
=====  
ACL handle : 0xe0000092  
ACL flags : 0x00000001  
Number of ACEs : 4  
Ace handle [1] : 0xf500010c  
Ace handle [2] : 0xd800010d  
Ace handle [3] : 0x4c00010e  
Ace handle [4] : 0x0600010f
```

```
ACL number : 2  
=====  
ACL handle : 0x14000093  
ACL flags : 0x00000001  
Number of ACEs : 1  
Ace handle [1] : 0x8e000110
```

Interface(s):  
Vlan10

```
#####  
#####  
##### Policy instance information #####  
#####  
#####  
Policy intf handle : 0xac000002  
Policy handle : 0x31000091
```

ID : 530 <-- 530 is the acl group ID

Protocol : [3] IPV4  
Feature : [23] AAL\_FEATURE\_VACL

Direction : [2] Egress <-- Egress VACL direction

Number of ACLs : 2  
Number of VMRs : 4-----

確認VACL工作正常。

- 故障排除與PACL和RACL部分相同。有關ping測試的詳細資訊，請參閱以下各節。
- 從10.1.1.3對10.1.1.2執行ping操作被應用的ACL策略拒絕。
- 檢查platform drop命令。

```
<#root>
```

```
9500H#
```

```
show platform software fed active acl counters hardware | inc VACL Drop
```

```
Ingress IPv4 VACL Drop
```

```
(0x23000006):
```

```
1011 frames      <-- Hardware level command displays drops against VACL
```

```
<...snip...>
```

## 案例 5. 群組/使用者端ACL(DACL)

組/客戶端ACL根據使用者組或客戶端的身分動態應用到使用者組。這些有時也稱為DAACL。

- 安全邊界：客戶端 ( 客戶端介面級別 )
- 附件：每個客戶端介面
- Direction：僅輸入
- 支援的ACL型別：MAC ACL和IP ACL ( 標準或擴展 )

## 配置GACL

```
<#root>
```

```
Cat9400#
```

```
show run interface gigabitEthernet 2/0/1
```

```
Building configuration...
```

```
Current configuration : 419 bytes
```

```
!
```

```
interface GigabitEthernet2/0/1
 switchport access vlan 10
 switchport mode access
 switchport voice vlan 5
```

```
ip access-group ACL-ALLOW in
```

```
<-- This is the pre-authenticated ACL (deny ip any any)
```

```
authentication periodic
authentication timer reauthenticate server
```



```
access-session control-direction in
access-session port-control auto
no snmp trap link-status
mab
dot1x pae authenticator
spanning-tree portfast
```

```
service-policy type control subscriber ISE_Gi2/0/1
```

```
end
```

```
Cat9400#
```

```
show access-session interface gigabitEthernet 2/0/1 details
```

```
Interface: GigabitEthernet2/0/1
```

```
IIF-ID: 0x1765EB2C <-- The IF_ID used in this example is dynamic
```

```
MAC Address: 000a.aaaa.aaaa <-- The client MAC
```

```
IPv6 Address: Unknown
IPv4 Address: 10.10.10.10
User-Name: 00-0A-AA-AA-AA-AA
```

```
Status: Authorized <-- Authorized client
```

```
Domain: VOICE
Oper host mode: multi-auth
Oper control dir: in
Session timeout: 300s (server), Remaining: 182s
Timeout action: Reauthenticate
Common Session ID: 27B17A0A000003F499620261
Acct Session ID: 0x000003e7
Handle: 0x590003ea
Current Policy: ISE_Gi2/0/1
```

```
Server Policies:
```

```
ACS ACL:
```

```
xACSACLx-IP-MAB-FULL-ACCESS-59fb6e5e
```

```
<-- The ACL pushed from ISE server
```

```
Method status list:
```

```
Method      State
dot1x      Stopped
```

```
mab          Authc Success
```

```
<-- Authenticated via MAB (Mac authenticator)
```

```
Cat9400#
```

```
show ip access-lists xACSACLx-IP-MAB-FULL-ACCESS-59fb6e5e
```

```
Extended IP access list xACSACLx-IP-MAB-FULL-ACCESS-GOOD-59fb6e5e
```

```
1 permit ip any any
```

```
<-- ISE pushed a permit ip any any
```

## 檢驗GACL

繫結到iif-id的組CG ID。

```
<#root>
```

```
Cat9400#
```

```
show platform software fed active acl interface 0x1765EB2C
```

```
<-- The IF_ID from the access
```

```
#####
#####
##### Printing Interface Infos #####
#####
#####
```

```
INTERFACE: Client MAC
```

```
000a.aaaa.aaaa
```

```
<-- Client MAC matches the access-session output
```

```
MAC
```

```
000a.aaaa.aaaa
```

```
#####
intfinfo: 0x7f104820cae8
Interface handle: 0x5a000110
```

```
Interface Type: Group
```

```
<-- This is a group ident
```

```
IIF ID: 0x1765eb2c
```

```
Input IPv4: Policy Handle: 0x9d00011e
```

```
Policy Name: ACL-ALLOW:xACSACLx-IP-MAB-FULL-ACCESS-59fb6e5e
```

```
:
```

```
<-- DACL name matches
```

```
CG ID: 127760
```

```
<-- The ACL group ID
```

```
CGM Feature: [35]
```

```
acl-grp
```

```
Bind Order: 0
```

與組GC ID關聯的ACL資訊。

<#root>

Cat9400#

show platform software fed active acl info acl-grp-cgid 127760

<-- the CG ID

#####
#####
##### Printing CG Entries #####
#####
#####

ACL CG (

acl-grp/127760

):

ACL-ALLOW:xACSACLx-IP-MAB-FULL-ACCESS-59fb6e5e

: type: IPv4

<-- Group ID & ACL name are correct

Total Ref count 1

1 CGACL

<-- 1

region reg\_id: 1
subregion subr\_id: 0
GCE#:1 #flds: 2 14:N matchall:N deny:N
Result: 0x04000000

ipv4\_src: value = 0x00000000, mask = 0x00000000
ipv4\_dst: value = 0x00000000, mask = 0x00000000

<-- Permits 1

GCE#:10 #flds: 2 14:N matchall:N deny:N
Result: 0x04000000
ipv4\_src: value = 0x00000000, mask = 0x00000000
ipv4\_dst: value = 0x00000000, mask = 0x00000000

案例 6.ACL記錄

裝置軟體可以提供有關標準IP訪問清單允許或拒絕的資料包的系統日誌消息。任何與ACL匹配的資料包都會導致有關資料包的資訊日誌消息傳送到控制檯。記錄到控制檯的消息級別由日誌控制檯控制系統日誌消息的命令。

- 與單播反向路徑轉發(uRPF)一起使用的ACL不支援ACL日誌消息。僅支援RACL。
• 從裝置控制平面生成的資料包不支援輸出方向的ACL日誌。
• 路由在硬體中完成，並登入軟體，因此，如果大量資料包與包含logkeyword的permit或deny

ACE匹配，則軟體無法與硬體處理速率匹配，並且無法記錄所有資料包。

- 觸發ACL的第一個資料包會立即生成日誌消息，後續資料包將在出現或記錄之前以5分鐘為間隔收集。該日誌消息包括訪問清單編號、資料包是被允許還是被拒絕、資料包的源IP地址以及在前5分鐘間隔內該源允許或拒絕的資料包數。
- 有關ACL日誌行為和限制的完整詳細資訊，請參閱相關資訊部分中說明的相應的安全配置指南Cisco IOS XE。

日誌示例PACL:

此範例顯示否定情況，其中ACL type和log關鍵字不能一起使用。

```
<#root>
9500H#
show access-lists TEST

Extended IP access list TEST
 10 permit ip host 10.1.1.1 any
log                <-- Log keyword applied to ACE entry

                20 deny ip host 10.1.1.3 any
log

9500H(config)#
interface twentyFiveGigE 1/0/1
9500H(config-if)#
ip access-group TEST in                <-- apply logged ACL
Switch Port ACLs are not supported for LOG!                <-- message indicates this is an unsupported combinat
```

日誌示例RAACL ( 拒絕 ) :

```
<#root>
9500H#
show access-lists TEST

Extended IP access list TEST
 10 permit ip host 10.1.1.1 any
log                <-- Log keyword applied to ACE entry

                20 deny ip host 10.1.1.3 any
log
```

```
9500H(config)#
```

```
interface vlan 10
```

```
9500H(config-if)#
```

```
ip access-group TEST in          <-- ACL applied to SVI
```

```
### Originate ICMP from 10.1.1.3 to 10.1.1.2 (denied by ACE) ###
```

```
C9300#
```

```
ping 10.1.1.2 source vlan 10 repeat 110
```

```
Type escape sequence to abort.
```

```
Sending 10, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
```

```
Packet sent with a source address of 10.1.1.3
```

```
.....
```

```
Success rate is 0 percent (0/110)
```

```
9500H#
```

```
show access-list TEST
```

```
Extended IP access list TEST
```

```
10 permit ip host 10.1.1.1 any log
```

```
20 deny ip host 10.1.1.3 any log (110 matches) <-- Matches increment in show access-list command
```

```
9500H#
```

```
show platform software fed active acl counters hardware | inc RACL
```

```
Ingress IPv4 RACL Drop (0xed000007): 0 frames
```

```
Ingress IPv4 RACL Drop and Log (0x93000009): 110 frames <-- Aggregate command shows hits on
```

```
%SEC-6-IPACCESSLOGDP: list TEST denied icmp 10.1.1.3 -> 10.1.1.2 (8/0), 10 packets <-- Syslog message i
```

日誌示例RACL(Permit):

當log語句用於permit語句時，軟體計數器的命中數顯示傳送的資料包數翻倍。

```
<#root>
```

```
C9300#
```

```
ping 10.1.1.2 source vlan 10 repeat 5          <-- 5 ICMP Requests are sent
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
```

```
Packet sent with a source address of 10.1.1.1
```

!!!!

Success rate is 100 percent (5/5)

, round-trip min/avg/max = 1/1/1 ms

9500H#

show access-lists TEST

Extended IP access list TEST

10 permit ip host 10.1.1.1 any log (10 matches) <-- Hit counter shows 10

20 deny ip host 10.1.1.3 any log (115 matches)

## 疑難排解

### ACL統計資訊

排除ACL故障時，必須瞭解裝置如何測量ACL統計資訊及其位置。

- ACL統計資訊在聚合級別收集，而不是在每個ACE級別收集。
- 硬體不能允許每個ACE或每個ACL統計資訊。
- Deny、Log和CPU轉發資料包等統計資訊將被收集。
- MAC、IPv4和IPv6資料包的統計資訊是單獨收集的。
- show platform software fed switch active acl counters hardware 可用於顯示聚集統計資訊。

### 清除ACL統計資訊

排查ACL問題時，清除各種ACL計數器以便獲得新的基線計數會有所幫助。

- 使用這些命令可以清除軟體和硬體ACL計數器統計資訊。
- 對ACL匹配/命中事件進行故障排除時，建議清除相關ACL以找到最新或相關的基線匹配。

<#root>

```
clear platform software fed active acl counters hardware
```

(clears the hardware matched counters)

```
clear ip access-list counters
```

(clears the software matched counters - IPv4)

```
clear ipv6 access-list counters
```

(clears the software matched counters - IPv6)

## ACL TCAM用完後會發生什麼情況？

- ACL始終應用於硬體TCAM。如果先前配置的ACL已使用TCAM，則新ACL無法獲得程式設計所需的所需ACL資源。
- 如果在耗盡TCAM後新增ACL，則會丟棄它所連線的介面的所有資料包。
- 在軟體中保留ACL的操作稱為解除安裝。
- 當資源可用時，交換機會自動嘗試將ACL程式設計到硬體中。如果成功，ACL會被推送到硬體，資料包開始轉發。
- 將軟體保留的ACL程式設計到TCAM中的操作稱為Reloading。
- PAACL、VAACL、RAACL和GAACL可以彼此獨立地解除安裝/重新載入。

## ACL TCAM耗盡

- 應用新增ACL的介面會在硬體資源可用之前開始捨棄封包。
- GAACL使用者端會進入UnAuth狀態。

## VCU耗盡

- 一旦超過L4OP限制或超過VCU，軟體將執行ACL擴展並建立新的ACE條目，以便在不使用VCU的情況下執行等效操作。
- 一旦發生這種情況，TCAM可能會從這些新增的條目中耗盡。

## ACL系統日誌錯誤

如果特定安全ACL資源用完，系統就會生成SYSLOG消息（介面、VLAN、標籤等，值可以不同）。

ACL日誌消息	定義	恢復操作
%ACL_ERRMSG-4-UNLOADED：已饋送交換機1：介面<interface>上的輸入<ACL>未程式設計到硬體中，並且流量被丟棄。	ACL已解除安裝（保留在軟體中）	研究TCAM規模。如果超出規模，請重新設計ACL。
%ACL_ERRMSG-6-REMOVED: 1已填充：已針對標籤<label>asic<number>刪除介面	解除安裝的ACL配置將從介	ACL已刪除，無需執行任何操作

<interface>上輸入<ACL>的已解除安裝配置。	面刪除	
%ACL_ERRMSG-6-RELOADED: 1已補給：介面<interface>上的輸入<ACL>現在已載入到asic<number>上標籤<label>的硬體中。	ACL現在已安裝在硬體中	ACL的問題現在已在硬體中解決，無需執行任何操作
%ACL_ERRMSG-3-ERROR: 1 fed：未按繫結順序<number>在<interface>上應用輸入<ACL> IP ACL <NAME>配置。	其他型別的ACL錯誤（例如dot1x ACL安裝失敗）	確認是否支援ACL配置，以及TCAM是否超出規模
%ACL_ERRMSG-6-GACL_INFO：交換機1的R0/0：已提供：GACL不支援日誌記錄。	GACL配置了日誌選項	GACL不支援日誌。從GACL中刪除日誌語句。
%ACL_ERRMSG-6-PACL_INFO：交換機1的R0/0：饋送：PAACL不支援日誌記錄。	PAACL配置了日誌選項	PAACL不支援日誌。從PAACL中刪除日誌語句。
%ACL_ERRMSG-3-ERROR：交換機1 R0/0：已饋送：輸入IPv4組ACL隱式拒絕：<name>：配置未應用於客戶端MAC 0000.0000.0000。	(dot1x)ACL無法應用到目標埠	確認是否支援ACL配置，以及TCAM是否超出規模

## 資源外情形和恢復操作

案例 1.ACL繫結	恢復操作
<ul style="list-style-type: none"> <li>ACL被建立並應用於介面或VLAN。</li> <li>由於「資源不足」情況（例如TCAM耗盡），繫結失敗。</li> <li>ACL中沒有ACE可以程式設計到TCAM中。ACL仍處於UNLOADED狀態。</li> <li>在UNLOADED狀態下，所有流量（包括控制資料包）都會在介面上丟棄，直到問題得到解決。</li> </ul>	重新設計ACL以降低TCAM的利用率。
案例 2.ACL編輯	恢復操作
<ul style="list-style-type: none"> <li>系統會建立一個ACL並將其應用到介面，並且會向此ACL中新增更多ACE條目，同時將這些</li> </ul>	重新設計ACL以降低TCAM的利用率。



<p>條目應用到介面。</p> <ul style="list-style-type: none"> <li>• 如果TCAM沒有資源，則編輯操作失敗。</li> <li>• ACL中沒有ACE可以程式設計到TCAM中。ACL仍然處於UNLOADED狀態。</li> <li>• 在UNLOADED狀態中，所有流量（包括控制封包）都會捨棄介面上的流量，直到問題解決為止。</li> <li>• 現有的ACL專案也會在UNLOADED狀態下失敗，直到解決此問題。</li> </ul>	
<p>案例 3.ACL重新繫結</p>	<p>恢復操作</p>
<ul style="list-style-type: none"> <li>• ACL Re-bind是將ACL附加到介面，然後將另一個ACL附加到同一介面而不分離第一個ACL的操作。</li> <li>• 第一個ACL已建立並成功連線。</li> <li>• 會建立一個名稱不同且通訊協定(IPv4/IPv6)相同的更大ACL，並將其連線到同一個介面。</li> <li>• 裝置成功分離第一個ACL並嘗試將新ACL附加到此介面。</li> <li>• 如果TCAM沒有資源，重新繫結操作將失敗。</li> <li>• ACL中沒有ACE可以程式設計到TCAM中。ACL仍處於UNLOADED狀態。</li> <li>• 在UNLOADED狀態下，所有流量（包括控制資料包）都會在介面上丟棄，直到問題得到解決。</li> </ul>	<p>重新設計ACL以降低TCAM的利用率。</p>
<p>案例 4.繫結空(Null)ACL</p>	<p>恢復操作</p>
<ul style="list-style-type: none"> <li>• 建立沒有ACE項的ACL並將其附加到介面。</li> <li>• 系統使用允許「任何ACE」在內部建立此ACL，並將其連線到硬體中的介面（在此狀態下允許所有流量）。</li> <li>• 然後，使用相同的名稱或編號將ACE條目新增到ACL中。新增每個ACE時，系統會對TCAM進行程式設計。</li> <li>• 如果在新增ACE條目時TCAM資源耗盡，則ACL將移至UNLOADED狀態。</li> <li>• 在UNLOADED狀態下，所有流量（包括控制資料包）都會在介面上丟棄，直到問題得到解決。</li> <li>• 現有的ACL專案也會在UNLOADED狀態下失敗，直到解決此問題。</li> </ul>	<p>重新設計ACL以降低TCAM的利用率。</p>

## 檢驗ACL規模

本節介紹用於確定ACL規模和TCAM利用率的命令。

FMAN存取清單摘要：

確定已配置的ACL和每個ACL的ACE總數。

```
<#root>
```

```
9500H#
```

```
show platform software access-list f0 summary
```

```
Access-list
```

	Index	Num	Ref	
Num ACEs				
-----				
TEST				
	1	1		2
<-- ACL TEST contains 2 ACE entries				
ELSE		2	1	1
DENY		3	0	1

ACL用法：

```
<#root>
```

```
9500H#
```

```
show platform software fed active acl usage
```

```
#####  
#####  
##### Printing Usage Infos #####  
#####  
#####  
#####
```

```
ACE Software VMR max:196608 used:283
```

```
<-- Value/Mask/Result entry usage
```

```
#####
```

```
=====
```

```
Feature Type
```

ACL Type

Dir

Name

Entries Used

VACL                      IPV4                      Ingress                      VACL                      4

<-- Type of ACL Feature, type of ACL, Direction ACL applied, name of ACL, and number of TCAM entries con

```

=====
Feature Type                      ACL Type                      Dir                      Name                      Entries Used
RACL                      IPV4                      Ingress                      TEST                      5

```

TCAM使用情況(17.x):

TCAM usage命令在16.x和17.x系列之間存在顯著差異。

<#root>

9500H#

show platform hardware fed active fwd-asic resource tcam utilization

Codes: EM - Exact\_Match,

I - Input

,

O - Output

, IO - Input & Output, NA - Not Applicable

CAM Utilization for ASIC [0]

Table                      Subtype

Dir

Max

Used

%Used

V4                      V6                      MPLS                      Other

-----

```
Security ACL Ipv4
```

```
TCAM
```

```
I
```

```
7168
```

```
16
```

```
0.22%
```

```
16      0      0      0
Security ACL Non Ipv4 TCAM I      5120      76      1.48%      0      36      0      40
Security ACL Ipv4      TCAM
```

```
o
```

```
7168      18      0.25%      18      0      0      0
Security ACL Non Ipv4 TCAM      0      8192      27      0.33%      0      22      0      5
```

```
<...snip...>
```

```
<-- Percentage used and other counters about ACL consumption
```

```
<-- Dir = ACL direction (Input/Output ACL)
```

TCAM使用情況(16.x):

TCAM usage命令在16.x和17.x系列之間存在顯著差異。

```
<#root>
```

```
C9300#
```

```
show platform hardware fed switch active fwd-asic resource tcam utilization
```

```
CAM Utilization for ASIC [0]
```

```
Table
```

```
Max Values
```

```
Used Values
```

```
-----
```

```
Security Access Control Entries
```

```
5120
```

```
126      <-- Total used of the Maximum
```

```
<...snip...>
```

自定義SDM模板 ( TCAM重新分配 )

使用Cisco IOS XE Bengaluru 17.4.1, 您可以使用SDM模板 `sdm prefer custom acl` 指令。

有關如何配置和驗證此功能的詳細資訊, 請參閱[系統管理配置指南\(Cisco IOS XE Bengaluru](#)

## [17.4.x \( Catalyst 9500交換機 \)。](#)

本節介紹一些基本配置和驗證。

驗證當前的SDM模板：

```
<#root>
9500H#
show sdm prefer

Showing SDM Template Info

This is the Core template.                                <-- Core SD

Security Ingress IPv4 Access Control Entries*:           7168 (current) - 7168 (proposed) <-- IPv4 AC

Security Ingress Non-IPv4 Access Control Entries*:       5120 (current) - 5120 (proposed)
Security Egress IPv4 Access Control Entries*:            7168 (current) - 7168 (proposed)
Security Egress Non-IPv4 Access Control Entries*:        8192 (current) - 8192 (proposed)

<...snip...>

9500H#
show sdm prefer custom user-input

Custom Template Feature Values are not modified

<-- No customization to SDM
```

修改當前的SDM模板：

- 9500H(config)#sdm優先使用自訂acl  
9500H(config-sdm-acl)#acl-ingress 26 priority 1 <— 應用新的26K值。(在配置指南中討論優先順序)  
9500H(config-sdm-acl)#acl-egress 20 priority 2  
9500H(config-sdm-acl)#exit  
使用 show sdm prefer custom 以便檢視建議的值和 sdm prefer custom commit 以便透過此CLI套用「view the changes」。
- 驗證對SDM配置檔案的更改。
- 9500H#show sdm prefer custom

顯示SDM模板資訊：

這是自定義模板及其詳細資訊。

入口安全訪問控制條目\*: 12288 (當前) — 26624 (建議) <— 當前和建議使用 (建議26K)  
出口安全訪問控制條目\*: 15360 (當前) — 20480 (建議)

9500H#show sdm prefer custom user-input

ACL功能使用者輸入

使用者輸入值

=====

功能名稱優先順序 規模

-----

入口安全訪問控制條目：1 26\*1024 <— 由使用者輸入修改為26 x 1024(26K)

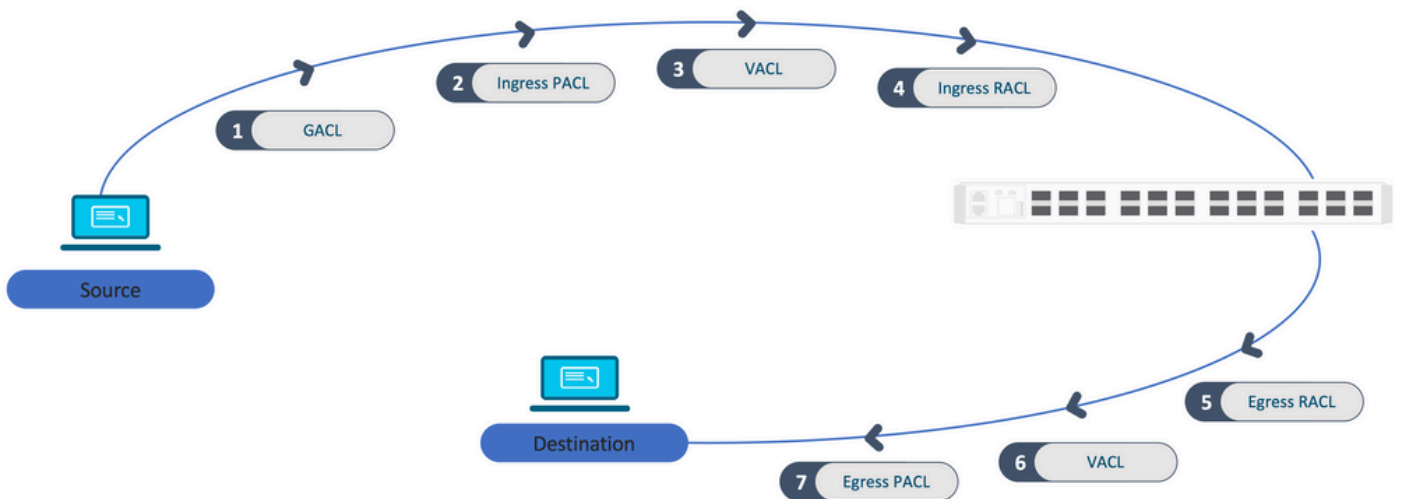
出口安全訪問控制項:2 20\*1024 <— 由使用者輸入修改為20 x 1024(20K)

- 將更改應用於SDM配置檔案。
- 9500H(config)#sdm首選自定義提交  
對正在運行的SDM首選項所做的更改將會儲存下來，並在下次重新載入時生效。 <— 重新載入後，ACL TCAM將分配給自定義值。

進一步閱讀：

ACL處理順序：

ACL的處理順序是從來源到目的地。



堆疊中程式化的ACL:

- 非連線埠型ACL ( 例如VACL、RAACL ) 會套用到任何交換器上的流量，而且會在堆疊中的所有交換器上程式化。
- 連線埠型ACL僅應用於連線埠上的流量，且僅對擁有介面的交換器進程式設計。
- ACL由活動交換機程式設計，隨後應用於成員交換機。
- 同樣的規則適用於其他冗餘選項，例如ISSU/SVL。

## ACL擴展：

- ACL擴展發生在裝置耗盡L4OP、Tables或VCU時。該裝置必須建立多個等價的ACE才能完成相同的邏輯，並且要快速耗盡TCAM。
- ### L4OP在規模上擴展，此ACL建立為##  
9500H(config)#ip access-list extended TEST  
9500H(config-ext-nacl)#permit tcp 10.0.0.0 0.255.255.255 any gt 150 <— 匹配埠151及更高版本

###必須將此擴展為多個不使用L4OP的ACE ###

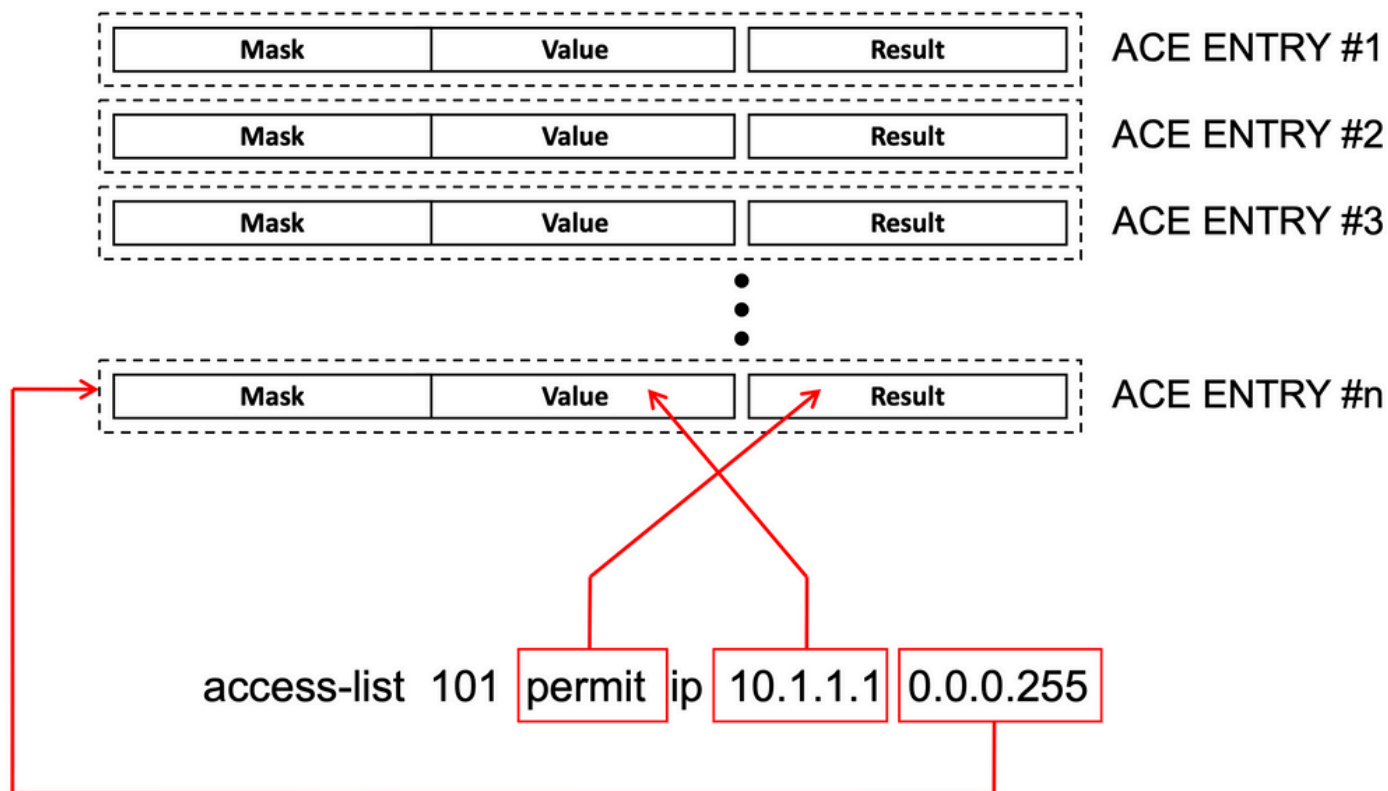
```
9500H(config-ext-nacl)#permit tcp 10.0.0.0 0.255.255.255 any eq 151
9500H(config-ext-nacl)#permit tcp 10.0.0.0 0.255.255.255 any eq 152
9500H(config-ext-nacl)#permit tcp 10.0.0.0 0.255.255.255 any eq 153
9500H(config-ext-nacl)#permit tcp 10.0.0.0 0.255.255.255 any eq 154
...等等....
```

## TCAM消費和標籤共用：

- 每個ACL策略都由標籤在內部引用。
- 將ACL策略（安全ACL，如GACL、PACL、VACL、RACL）應用到多個介面或VLAN時，它使用相同的標籤。
- 輸入/輸出ACL使用不同的標籤空間。
- IPv4、IPv6和MAC ACL使用其他標籤空間。
- 相同的PACL應用於介面A的輸入和介面A的輸出。TCAM中有兩個PACL例項，每個例項都有一個唯一的入口和出口標籤。
- 如果將具有L4OP的相同PACL應用於每個核心上存在的多個輸入介面，則在TCAM中程式設計的同—PACL有兩個例項，每個核心一個。

## VMR描述：

ACE在TCAM內部被程式設計為「VMR」，也稱為「值」、「掩碼」、「結果」。每個ACE條目均可使用VMR並且可以使用VCU。



ACL可擴充性：

安全ACL資源專用於安全ACL。它們不會與其他功能共用。

ACL TCAM資源	Cisco Catalyst 9600	Cisco Catalyst 9500	Cisco Catalyst 9400	Cisco Catalyst 9300	Cisco Catalyst 9200			
IPv4專 案	輸入 : 12000*	Egress: 15000*	C9500:18000*	C9500高 效能 輸入 : 12000* 出口 : 15000*	18000*	C9300: 5000	C9300B: 18000	C9300X:8000
IPv6條 目	IPv4條目的一半		IPv4條目的一半		IPv4條 目的一 半	IPv4條目的一半		
一種型 別的 IPv4	12000		C9500:18000	C9500高 效能: 15000	18000	C9300: 5000	C9300B : 配 18000	C9300X:8000



ACL條目不能超過							
一種型別的IPv6 ACL條目不能超過	6000	C9500: 9000	C9500高性能： 7500	9000	2500/9000/4000		
L4OP/標籤	8	8		8	8		
輸入VCU	192	192		192	192		
輸出VCU	96	96		96	96		

## 相關資訊

- [安全配置指南，Cisco IOS XE阿姆斯特丹版17.3.x \( Catalyst 9200交換機 \)](#)
- [安全配置指南，Cisco IOS XE阿姆斯特丹版17.3.x \( Catalyst 9300交換機 \)](#)
- [安全配置指南，Cisco IOS XE阿姆斯特丹版17.3.x \( Catalyst 9400交換機 \)](#)
- [安全配置指南，Cisco IOS XE阿姆斯特丹版17.3.x \( Catalyst 9500交換機 \)](#)
- [安全配置指南，Cisco IOS XE阿姆斯特丹版17.3.x \( Catalyst 9600交換機 \)](#)
- [系統管理配置指南，Cisco IOS XE Bengaluru 17.4.x \( Catalyst 9500交換機 \)](#)
- [思科技術支援與下載](#)

## Debug和Trace命令

編號	指令	備註
1	show platform hardware fed [switch] active fwd-asic drops exceptions asic <0>	轉儲ASIC伺服器上的異常計#N器。
2	show platform software fed [switch] active acl	此命令將列印該框中所有已配置ACL的相關資訊以及介面和策略資訊。

3	show platform software fed [switch] active acl policy 18	此命令只列印有關策略18的資訊。您可以從命令2獲取此策略ID。
4	show platform software fed [switch] active acl interface intftype pacl	此命令會根據介面型別 ( pacl/vacl/racl/gacl/sgacl等 ) 列印有關ACL的資訊。
5	show platform software fed [switch] active acl interface intftype pacl acltype ipv4	此命令會根據介面型別 ( pacl/vacl/racl/gacl/sgacl等 ) 列印有關ACL的資訊，並會按通訊協定進行過濾 ( ipv4/ipv6/mac等 )。
6	show platform software fed [switch] active acl interface intftype pacl acltype ipv4	此命令列印有關介面的資訊。
7	show platform software fed [switch] active acl interface 0x9	此命令根據IIF-ID ( 命令from 6 ) 列印在介面上應用的ACL的簡短資訊。
8	show platform software fed [switch] active acl definition	此命令將列印有關機箱上配置的ACL及其存在於CGD中的資訊。
9	show platform software fed [switch] active acl iifid 0x9	此命令根據IIF-ID列印介面上應用的ACL的詳細資訊。
10	show platform software fed [switch] active acl usage	此命令根據功能型別列印每個ACL使用的VMR數量。
11	show platform software fed [switch] active acl policy intftype pacl vcu	此命令會根據介面型別 ( pacl/vacl/racl/gacl/sgacl等 ) 為您提供原則資訊和VCU資訊。
12	show platform software fed [switch] active acl policy intftype pacl cam	此命令根據介面型別 ( pacl/vacl/racl/gacl/sgacl等 ) 為您提供有關CAM中VMR的策略資訊和詳細資訊。
13	show platform software interface [switch] [active] R0 brief	此命令為您提供了有關該框上介面的詳細資訊。
14	show platform software fed [switch] active port if_id 9	此命令根據IIF-ID列印埠的詳細資訊。
15	show platform software fed [switch] active vlan	此命令列印有關VLAN 30的詳細資訊。

	30	
16	show platform software fed [switch] active acl cam asic 0	此命令會在正在使用的ASIC 0上列印完整的ACL cam。
17	show platform software fed [switch] active acl counters hardware	此命令列印硬體中的所有ACL計數器。
18	show platform hardware fed [switch] active fwd- asic resource tcam table pbr record 0 format 0	在列印PBR部分的條目時，可以為ACL和CPP等不同部分指定而不是PBR。
19	show platform software fed [switch] active punt cpuq [1 2 3 ...]	為了檢查某個CPU隊列上的活動，您還可以選擇清除隊列狀態以進行調試。
20	show platform software fed [switch] active ifm mappings gpn	使用IIF-ID和GPN列印介面對映
21	show platform software fed [switch active ifm if- id	列印有關介面配置和與ASIC的關聯的資訊。此命令有助於檢查ASIC和CORE的介面。
22	set platform software trace fed [switch] active acl/asic_vmr/asic_vcu/cgac1/sgac1 [debug error ...]	設定FED中特定功能的跟蹤。
23	request platform software trace rotate all	正在清除跟蹤緩衝區。
24	show platform software trace message fed [switch] active	正在列印FED的跟蹤緩衝區。
25	set platform software trace forwarding-manager [switch] [active] f0 fman [debug error ...]	正在啟用FMAN跟蹤。
26	show platform software trace message forwarding-manager [switch] [active] f0	正在列印FMAN的跟蹤緩衝區。
27	debug platform software infrastructure punt detail	在PUNT上設定調試。

28	debug ip cef packet all input rate 100	CEF資料包調試已開啟。
----	--	--------------

## 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）。