驗證ACI交換矩陣中的合約和規則

目錄

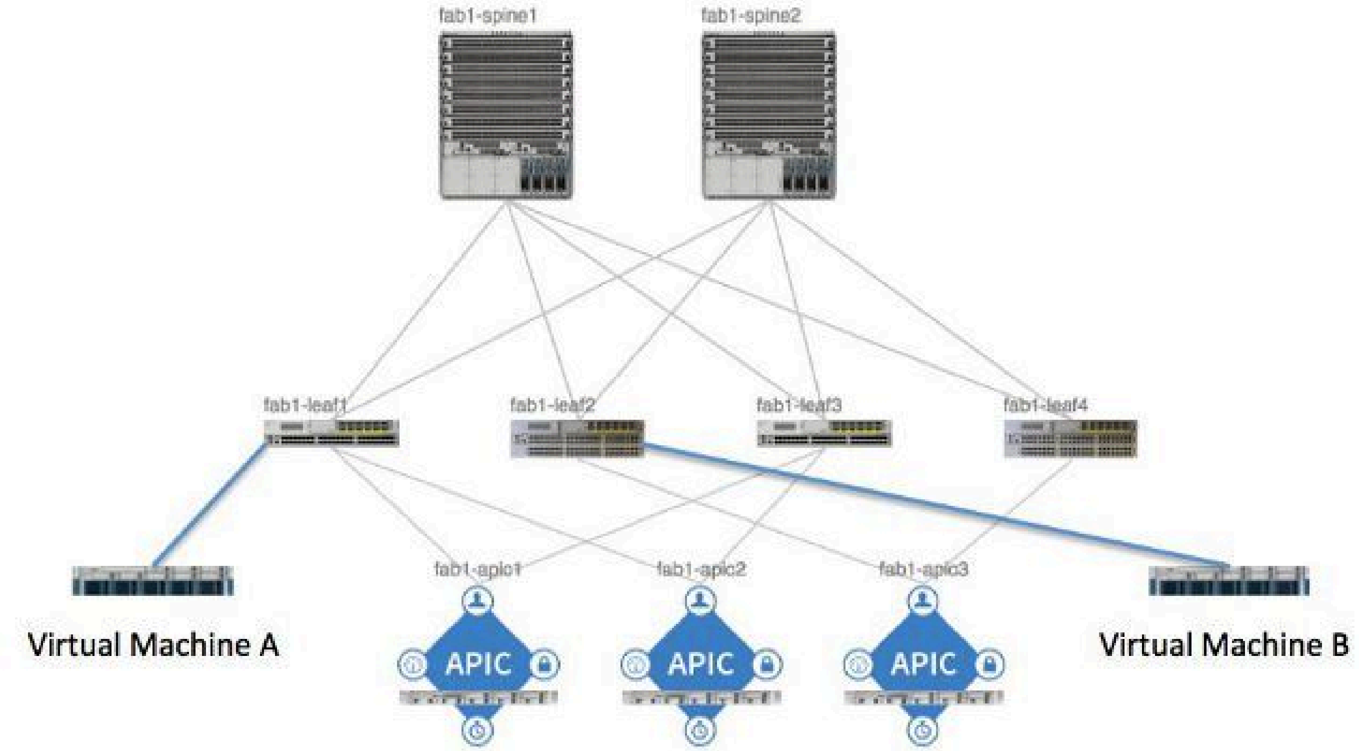
簡介

本文說明如何驗證在以應用為中心的基礎設施(ACI)交換矩陣中是否配置了合約以及合約是否行為正確。

拓撲

在本文檔中使用的示例中，虛擬機器A(VM)連線到Leaf1，並且存在允許其與VM-B（連線到Leaf2）進行通訊的合約。合約允許網際網路控制訊息通訊協定(ICMP)和HTTP。

此圖說明拓撲：



流程概述

這是合約和規則的策略互動和流程：

1. 應用策略基礎設施控制器(APIC)上的策略管理器與交換機上的策略元素管理器通訊。

2. 交換機上的策略元素管理器對交換機上的對象儲存進行程式設計。

3. 交換器上的原則管理員與交換器上的存取控制清單服務品質(ACLQOS)使用者端通訊。

4. ACLQOS客戶端對硬體進行程式設計。

確定使用的合約/分割槽規則

以下是為兩個終端組(EPG)新增約定之前，從枝葉輸出的show zoning-rule命令示例。

<#root>

```
fab1_leaf1#

show zoning-rule


Rule ID  SrcEPG  DstEPG  FilterID  operSt   Scope     Action

=======  ======  ======  ========  ======   =====     ======

4096     0       0       implicit  enabled  16777200  deny,log
```

```
4097     0       0       implicit  enabled  3080192   deny,log
4098     0       0       implicit  enabled  2686976   deny,log
4099     0       49154   implicit  enabled  2686976   permit
4102     0       0       implicit  enabled  2097152   deny,log
4103     0       32771   implicit  enabled  2097152   permit
4117     16387   16386   12        enabled  2097152   permit
4116     16386   16387   13        enabled  2097152   permit
4100     16386   49154   default   enabled  2097152   permit
4101     49154   16386   default   enabled  2097152   permit
4104     0       32770   implicit  enabled  2097152   permit
4105     49155   16387   13        enabled  2097152   permit
4112     16387   49155   13        enabled  2097152   permit
4113     49155   16387   12        enabled  2097152   permit
4114     16387   49155   12        enabled  2097152   permit
[snip]
```

這是新增合約後的相同命令輸出，以便兩個EPG可以相互通訊：

<#root>

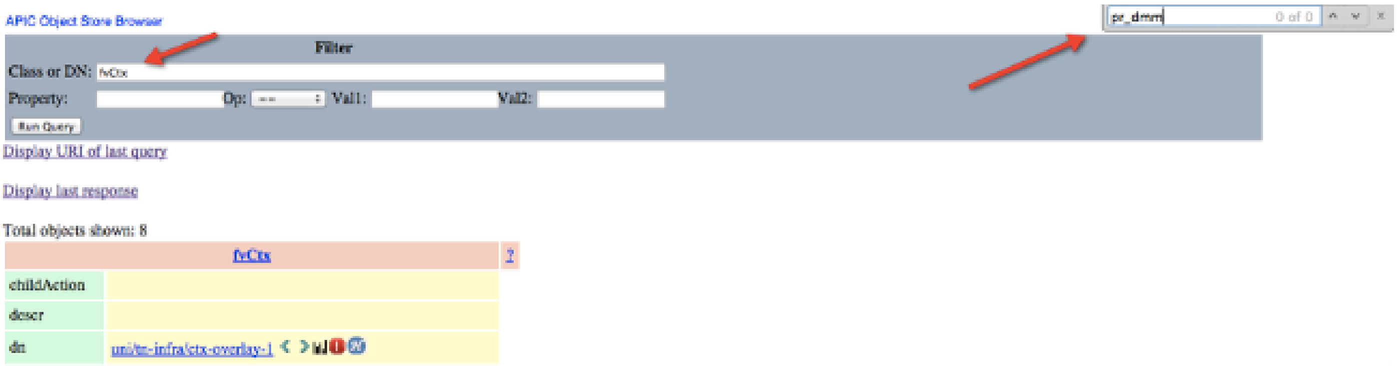
```
fab1_leaf1#
show zoning-rule

Rule ID  SrcEPG  DstEPG  FilterID  operSt   Scope     Action
=======  ======  ======  ========  ======   ========  ========
4096     0       0       implicit  enabled  16777200  deny,log
4097     0       0       implicit  enabled  3080192   deny,log
4098     0       0       implicit  enabled  2686976   deny,log
4099     0       49154   implicit  enabled  2686976   permit

4131     49155   32771   7         enabled  2686976   permit
```

```
4132      32771    49155    6           enabled   2686976    permit


4102      0        0        implicit    enabled   2097152    deny,log

4103      0        32771    implicit    enabled   2097152    permit

4117      16387    16386    12          enabled   2097152    permit

4116      16386    16387    13          enabled   2097152    permit

4100      16386    49154    default     enabled   2097152    permit

4101      49154    16386    default     enabled   2097152    permit

4104      0        32770    implicit    enabled   2097152    permit

4105      49155    16387    13          enabled   2097152    permit

4112      16387    49155    13          enabled   2097152    permit

4113      49155    16387    12          enabled   2097152    permit

4114      16387    49155    12          enabled   2097152    permit

[snip]
```

注意：請注意已新增的新規則ID（4131和4132）、7和6的篩選器ID以及ID范2686976。

注意：此命令輸出使您可以輕鬆找到必須在實驗室系統中檢查的規則；但是，在生產環境中如果發生動態更改，可能會非常麻煩。

查詢感興趣規則的另一種方法是使用Visore。對上下文管理對象(MO)搜尋fvCtx。然後，您可以在該螢幕上搜尋您的特定上下文可分辨名稱(DN)，如下所示：



Class or DN: fvCtx

Property: Op: == Val1: Val2:

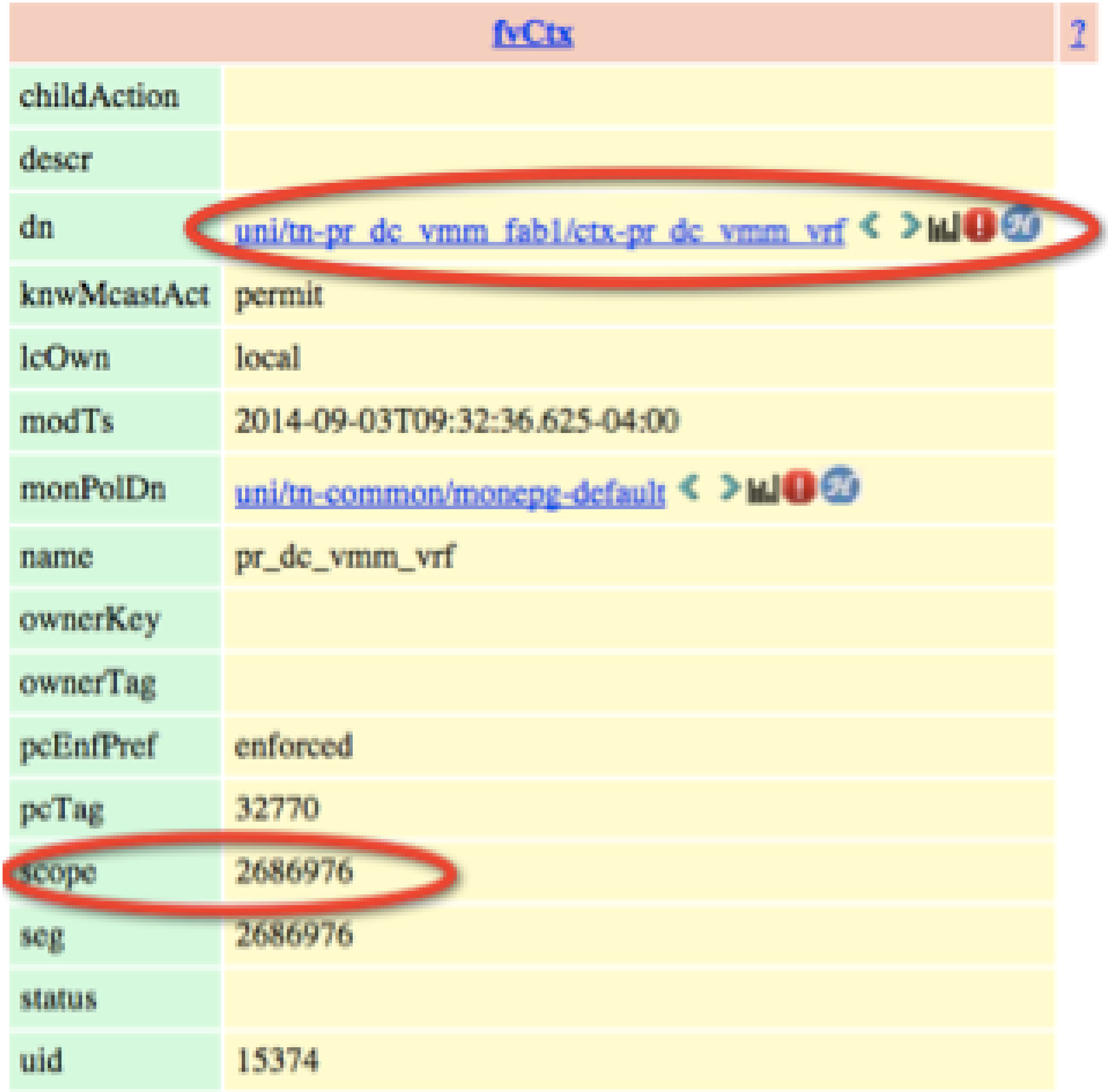
Display URI of last query

Display last response

Total objects shown: 8

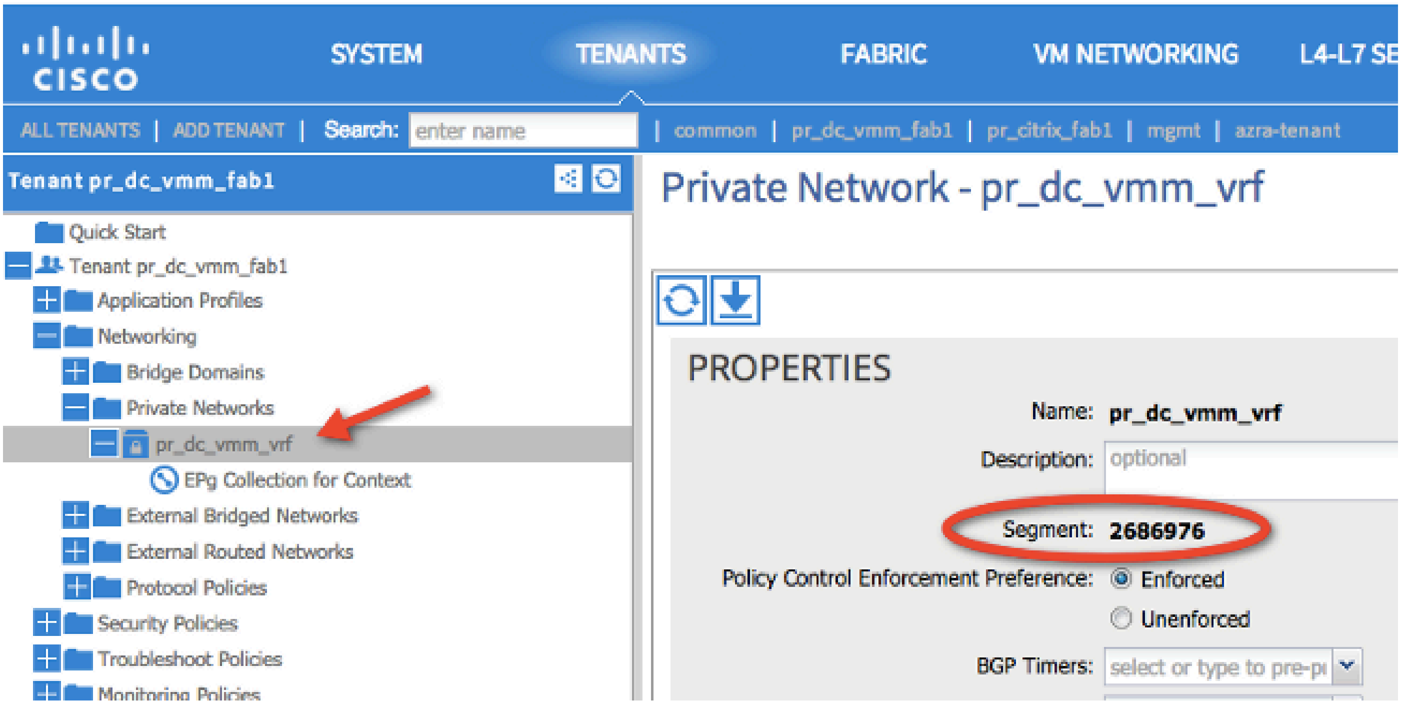
fvCtx	?
childAction	
descr	
dn	uni/tn-infra/ctx-overlay-1

請注意這一背景的範圍。您可以使用此命令對映到show-zoning-rule命令輸出，以便找到必須查詢的規則：



fvCtx	
childAction	
descr	
dn	uni/tn-pr_dc_vmm_fab1/ctx-pr_dc_vmm_vrf < >
knwMcastAct	permit
lcOwn	local
modTs	2014-09-03T09:32:36.625-04:00
monPolDn	uni/tn-common/monepg-default < >
name	pr_dc_vmm_vrf
ownerKey	
ownerTag	
pcEnfPref	enforced
pcTag	32770
scope	2686976
seg	2686976
status	
uid	15374

您還可以通過使用者介面(UI)標識上下文的段ID/範圍，如下所示：



Private Network - pr_dc_vmm_vrf

PROPERTIES

Name: pr_dc_vmm_vrf

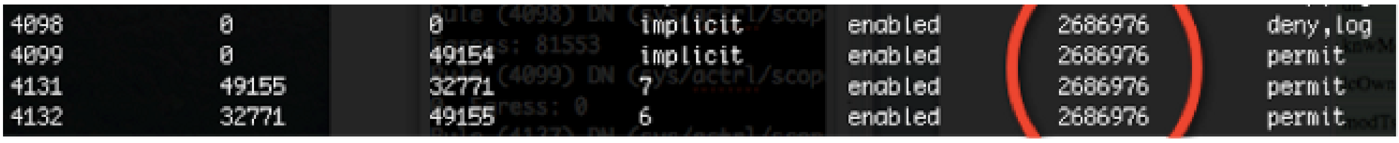
Description: optional

Segment: 2686976

Policy Control Enforcement Preference: ● Enforced ○ Unenforced

BGP Timers: select or type to pre-pı

此作用域與show zoning-rules命令輸出中顯示的內容相符：



```
4098        0        0        implicit   enabled   2686976   deny,log
4099        0        49154    implicit   enabled   2686976   permit
4131        49155    32771    7          enabled   2686976   permit
4132        32771    49155    6          enabled   2686976   permit
```

一旦您擁有作用域ID資訊並識別了規則和過濾器ID，就可以使用下一個命令來驗證您是否命中了新過濾器（而不是EPG之間的隱式拒絕消息）。包含隱含的deny訊息，因此預設情況下，EPG無法通訊。

請注意，在此命令輸出中，Leaf1、Filter-6(f-6)正在遞增：

<#root>

```
fab1_leaf1#

show system internal policy-mgr stats | grep 2686976



Rule (4098) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-any-f-implicit)
 Ingress: 0, Egress: 81553

Rule (4099) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-49154-f-implicit)
 Ingress: 0, Egress: 0

Rule (4131) DN (sys/actrl/scope-2686976/rule-2686976-s-49155-d-32771-f-7)
 Ingress: 0, Egress: 0


Rule (4132) DN (sys/actrl/scope-2686976/rule-2686976-s-32771-d-49155-f-6)
 Ingress: 1440, Egress: 0
```

<#root>

```
fab1_leaf1#

show system internal policy-mgr stats | grep 2686976



Rule (4098) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-any-f-implicit)
 Ingress: 0, Egress: 81553

Rule (4099) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-49154-f-implicit)
 Ingress: 0, Egress: 0

Rule (4131) DN (sys/actrl/scope-2686976/rule-2686976-s-49155-d-32771-f-7)
 Ingress: 0, Egress: 0


Rule (4132) DN (sys/actrl/scope-2686976/rule-2686976-s-32771-d-49155-f-6)



 Ingress: 1470, Egress: 0
```

請注意，在此命令輸出中，Leaf2、Filter-7(f-7)正在遞增：

<#root>

```
fab1_leaf2#

show system internal policy-mgr stats | grep 268697



Rule (4098) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-any-f-implicit)
 Ingress: 0, Egress: 80257

Rule (4099) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-49153-f-implicit)
 Ingress: 0, Egress: 0

Rule (4117) DN (sys/actrl/scope-2686976/rule-2686976-s-32771-d-49155-f-6)
 Ingress: 0, Egress: 0


Rule (4118) DN (sys/actrl/scope-2686976/rule-2686976-s-49155-d-32771-f-7)
 Ingress: 2481, Egress: 0
```

<#root>

```
fab1_leaf2#

show system internal policy-mgr stats | grep 268697



Rule (4098) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-any-f-implicit)
 Ingress: 0, Egress: 80257

Rule (4099) DN (sys/actrl/scope-2686976/rule-2686976-s-any-d-49153-f-implicit)
```

```
 Ingress: 0, Egress: 0

Rule (4117) DN (sys/actrl/scope-2686976/rule-2686976-s-32771-d-49155-f-6)
 Ingress: 0, Egress: 0


Rule (4118) DN (sys/actrl/scope-2686976/rule-2686976-s-49155-d-32771-f-7)
 Ingress: 2511, Egress: 0
```

提示：有關範圍、規則ID、目標、源pcTags和過濾器的知識對於進一步解決此問題的嘗試非常重要。瞭解存在規則ID的EPG也很有用。

您可以通過moquery命令使用DN名稱fvAEPg和grep對特定pcTag的MO執行搜尋，如下所示：

<#root>

```
admin@RTP_Apic1:~>

moquery -c fvAEPg | grep 49155 -B 5



dn : uni/tn-Prod/ap-commerceworkspace/

epg-Web


lcOwn : local
matchT : AtleastOne
modTs : 2014-10-16T01:27:35.355-04:00
monPolDn : uni/tn-common/monepg-default

pcTag : 49155
```

您還可以將filter選項與moquery命令一起使用，如下所示：

<#root>

```
admin@RTP_Apic1:~>

moquery -c fvAEPg -f 'fv.AEPg.pcTag=="49155"'


Total Objects shown: 1

# fv.AEPg
name : Web
childAction :
configIssues :
configSt : applied
descr :
dn : uni/tn-Prod/ap-commerceworkspace/

epg-Web
```

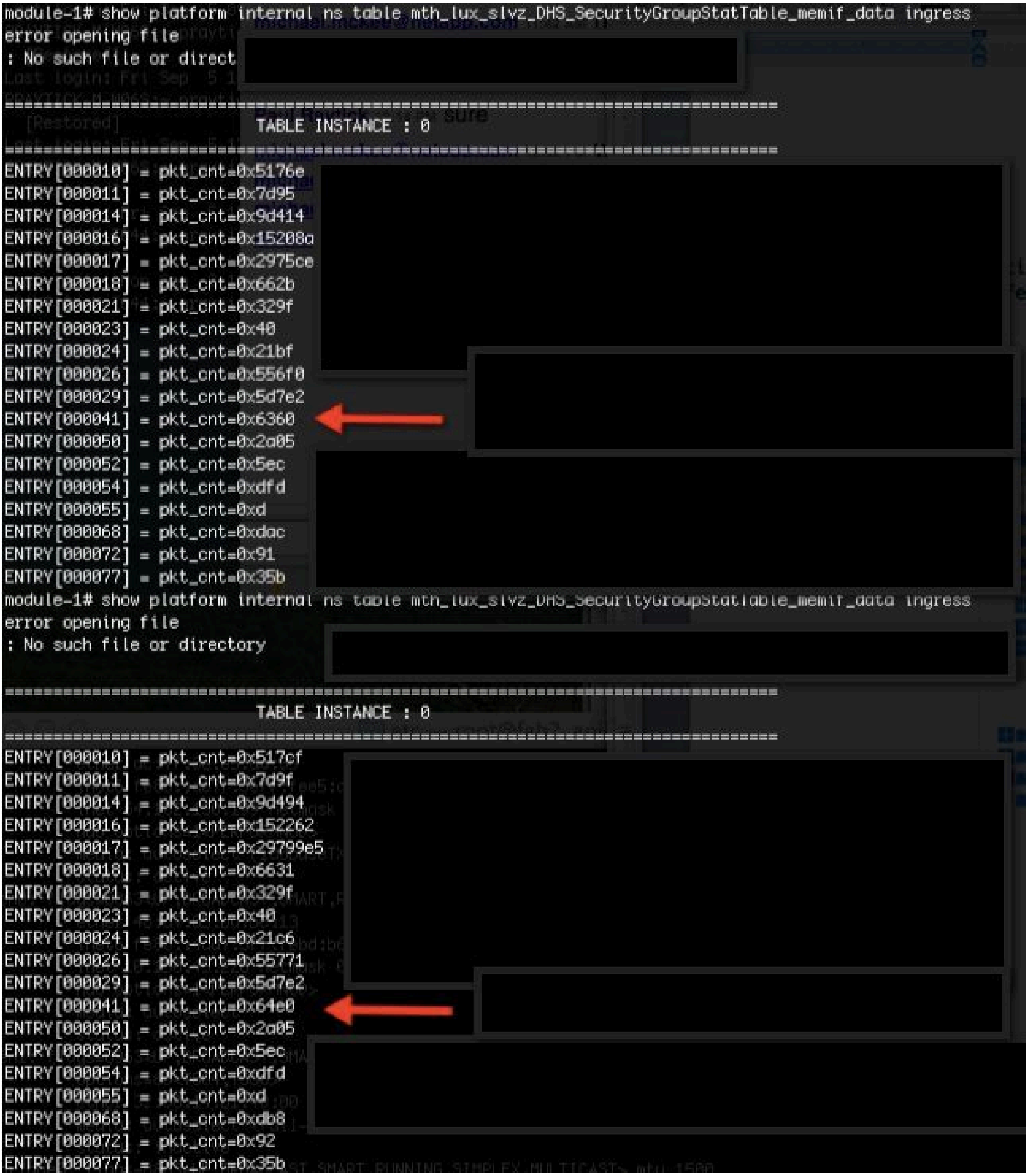
```
lcOwn : local
matchT : AtleastOne
modTs : 2014-10-16T01:27:35.355-04:00
monPolDn : uni/tn-common/monepg-default
```

```
pcTag : 49155
```

```
prio : unspecified
rn : epg-Web
scope : 2523136
status :
triggerSt : triggerable
uid : 15374
```

驗證硬體程式設計

現在，您可以驗證規則的硬體條目。要檢視硬體資訊，請輸入show platform internal ns table mth_lux_slvz_DHS_SecurityGroupStatTable_memif_data ingress命令(這是vsh_lc命令):



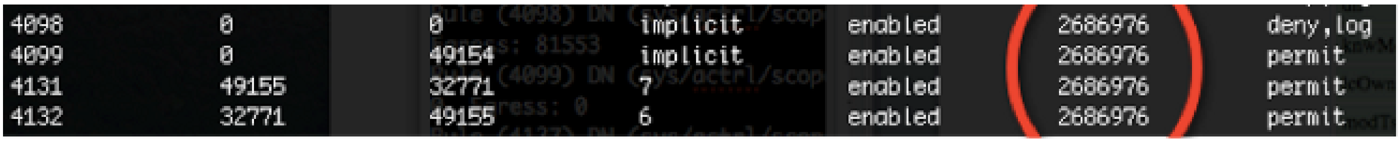
```
module-1# show platform internal ns table mth_lux_slvz_DHS_SecurityGroupStatTable_memif_data ingress
error opening file
: No such file or direct [redacted]

================================================================================
                          TABLE INSTANCE : 0
================================================================================
ENTRY[000010] = pkt_cnt=0x5176e [redacted]
ENTRY[000011] = pkt_cnt=0x7d95
ENTRY[000014] = pkt_cnt=0x9d414
ENTRY[000016] = pkt_cnt=0x15208a
ENTRY[000017] = pkt_cnt=0x2975ce
ENTRY[000018] = pkt_cnt=0x662b
ENTRY[000021] = pkt_cnt=0x329f
ENTRY[000023] = pkt_cnt=0x40
ENTRY[000024] = pkt_cnt=0x21bf
ENTRY[000026] = pkt_cnt=0x556f0 [redacted]
ENTRY[000029] = pkt_cnt=0x5d7e2
ENTRY[000041] = pkt_cnt=0x6360
ENTRY[000050] = pkt_cnt=0x2a05
ENTRY[000052] = pkt_cnt=0x5ec [redacted]
ENTRY[000054] = pkt_cnt=0xdfd
ENTRY[000055] = pkt_cnt=0xd
ENTRY[000068] = pkt_cnt=0xdac
ENTRY[000072] = pkt_cnt=0x91
ENTRY[000077] = pkt_cnt=0x35b
module-1# show platform internal ns table mth_lux_slvz_DHS_SecurityGroupStatTable_memif_data ingress
error opening file
: No such file or directory [redacted]

================================================================================
                          TABLE INSTANCE : 0
================================================================================
ENTRY[000010] = pkt_cnt=0x517cf
ENTRY[000011] = pkt_cnt=0x7d9f [redacted]
ENTRY[000014] = pkt_cnt=0x9d494
ENTRY[000016] = pkt_cnt=0x152262
ENTRY[000017] = pkt_cnt=0x29799e5
ENTRY[000018] = pkt_cnt=0x6631
ENTRY[000021] = pkt_cnt=0x329f
ENTRY[000023] = pkt_cnt=0x40
ENTRY[000024] = pkt_cnt=0x21c6
ENTRY[000026] = pkt_cnt=0x55771
ENTRY[000029] = pkt_cnt=0x5d7e2 [redacted]
ENTRY[000041] = pkt_cnt=0x64e0
ENTRY[000050] = pkt_cnt=0x2a05
ENTRY[000052] = pkt_cnt=0x5ec [redacted]
ENTRY[000054] = pkt_cnt=0xdfd
ENTRY[000055] = pkt_cnt=0xd
ENTRY[000068] = pkt_cnt=0xdb8
ENTRY[000072] = pkt_cnt=0x92
ENTRY[000077] = pkt_cnt=0x35b
```

在本示例中，硬體條目41(條目[000041])正在遞增。

註：顯示的上一個命令用於Northstar ASIC。用於Donner或Donner+的命令是show platform internal ns table mth_luxh_slvy_DHS_SecurityGroupStatTable_memif_data。

注意：在生產環境中使用此命令並不實用，但您可以改用本節中介紹的其他命令。

記住規則(4132)和範圍(268976)。



```
4098    0        0       implicit  enabled   2686976   deny,log
4099    0        49154   implicit  enabled   2686976   permit
4131    49155    32771   7         enabled   2686976   permit
4132    32771    49155   6         enabled   2686976   permit
```

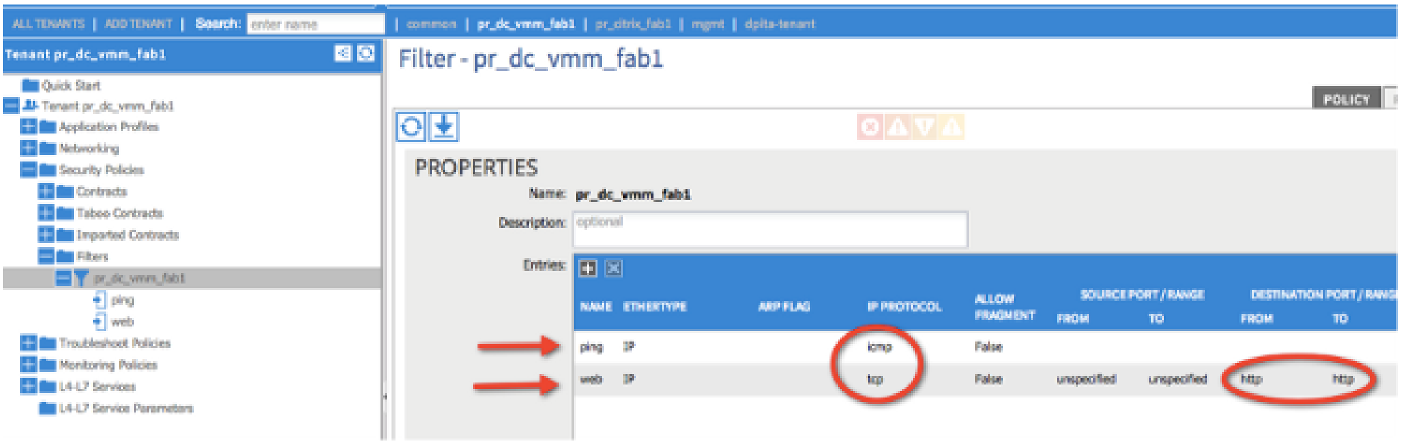
輸入以下命令可確定三重內容可定址儲存器(TCAM)硬體索引條目對映的規則ID，並根據規則ID和/或過濾器ID進行過濾：

<#root>

```
module-1#

show system internal aclqos zoning-rules



[snip]

===========================================
Rule ID: 4131 Scope 4 Src EPG: 49155 Dst EPG: 32771 Filter 7

Curr TCAM resource:
=============================
   unit_id: 0
   === Region priority: 771 (rule prio: 3 entry: 3)===
       sw_index = 62 |

hw_index = 40


   === Region priority: 772 (rule prio: 3 entry: 4)===
       sw_index = 63 |

hw_index = 45



===========================================
Rule ID: 4132 Scope 4 Src EPG: 32771 Dst EPG: 49155 Filter 6

Curr TCAM resource:
=============================
   unit_id: 0
   === Region priority: 771 (rule prio: 3 entry: 3)===
       sw_index = 66 |

hw_index = 41


   === Region priority: 771 (rule prio: 3 entry: 3)===
       sw_index = 67 |

hw_index = 42



[snip]
```

在本示例中，感興趣的源和目標EPG組合為32771=0x8003,49155=0xC003。因此，您可以考慮與規則ID（4131和4132）和篩選器ID（6和7）匹配的這些源類和目標類的所有TCAM條目。

在本例中，其中一些TCAM條目將被轉儲。例如，以下是允許這些EPG的ping和Web流量的合約設定：



Filter - pr_dc_vmm_fab1

PROPERTIES

Name: pr_dc_vmm_fab1

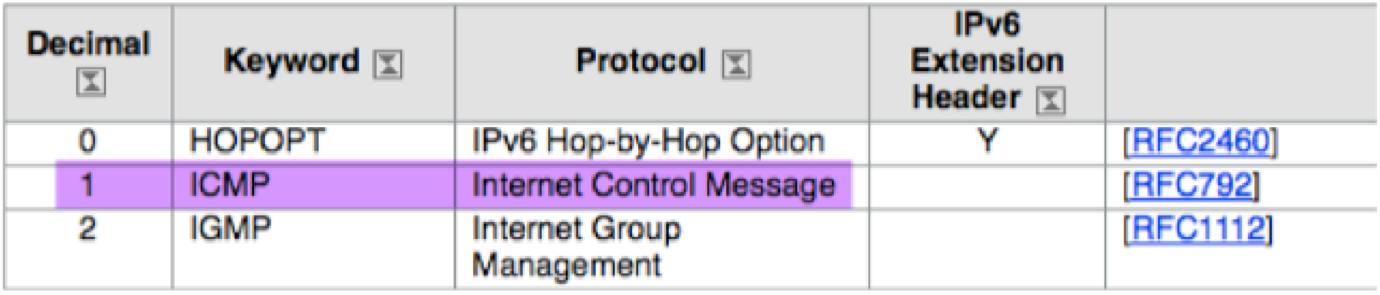
Description: optional

Entries:

NAME	ETHERTYPE	ARP FLAG	IP PROTOCOL	ALLOW FRAGMENT	SOURCE PORT / RANGE		DESTINATION PORT / RANG	
					FROM	TO	FROM	TO
ping	IP		icmp	False				
web	IP		tcp	False	unspecified	unspecified	http	http

```
<#root>

module-1#

show platform internal ns table mth_lux_slvz_DHS_SecurityGroupKeyTable0


 _memif_data 41



=======================================================================
                         TABLE INSTANCE : 0
=======================================================================
ENTRY[000041] =
               sg_label=0x4

               sclass=0x8003

               dclass=0xc003

               prot=0x1

 (IP Protocol 0x01 = ICMP)
```

註：顯示的上一個命令用於Northstar ASIC。用於Donner或Donner+的命令是show platform internal ns table mth_luxh_slvq_DHS_SecurityGroupKeyTable0_memif_data。



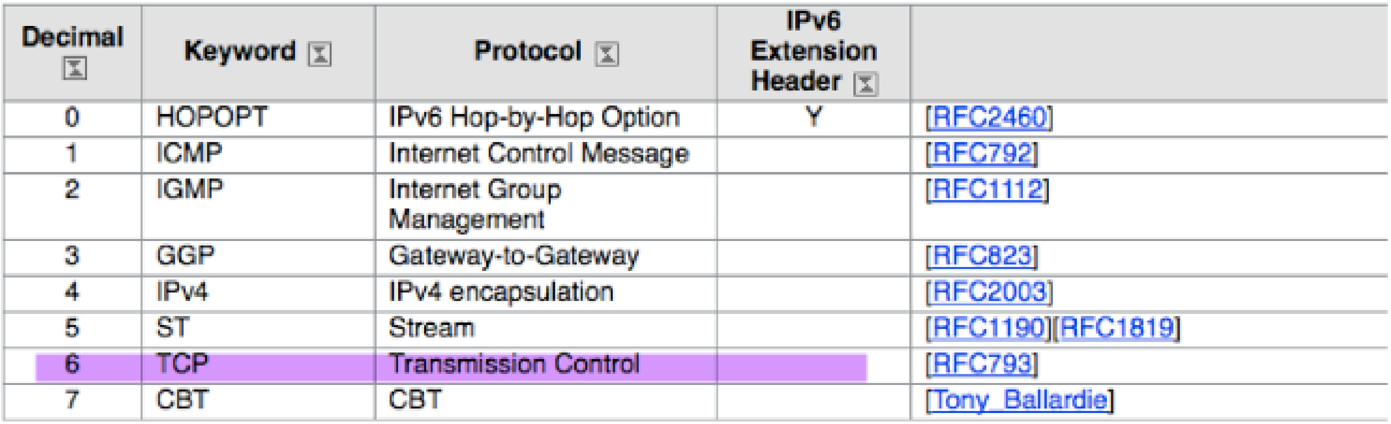
Decimal	Keyword	Protocol	IPv6 Extension Header	
0	HOPOPT	IPv6 Hop-by-Hop Option	Y	[RFC2460]
1	ICMP	Internet Control Message		[RFC792]
2	IGMP	Internet Group Management		[RFC1112]

<#root>

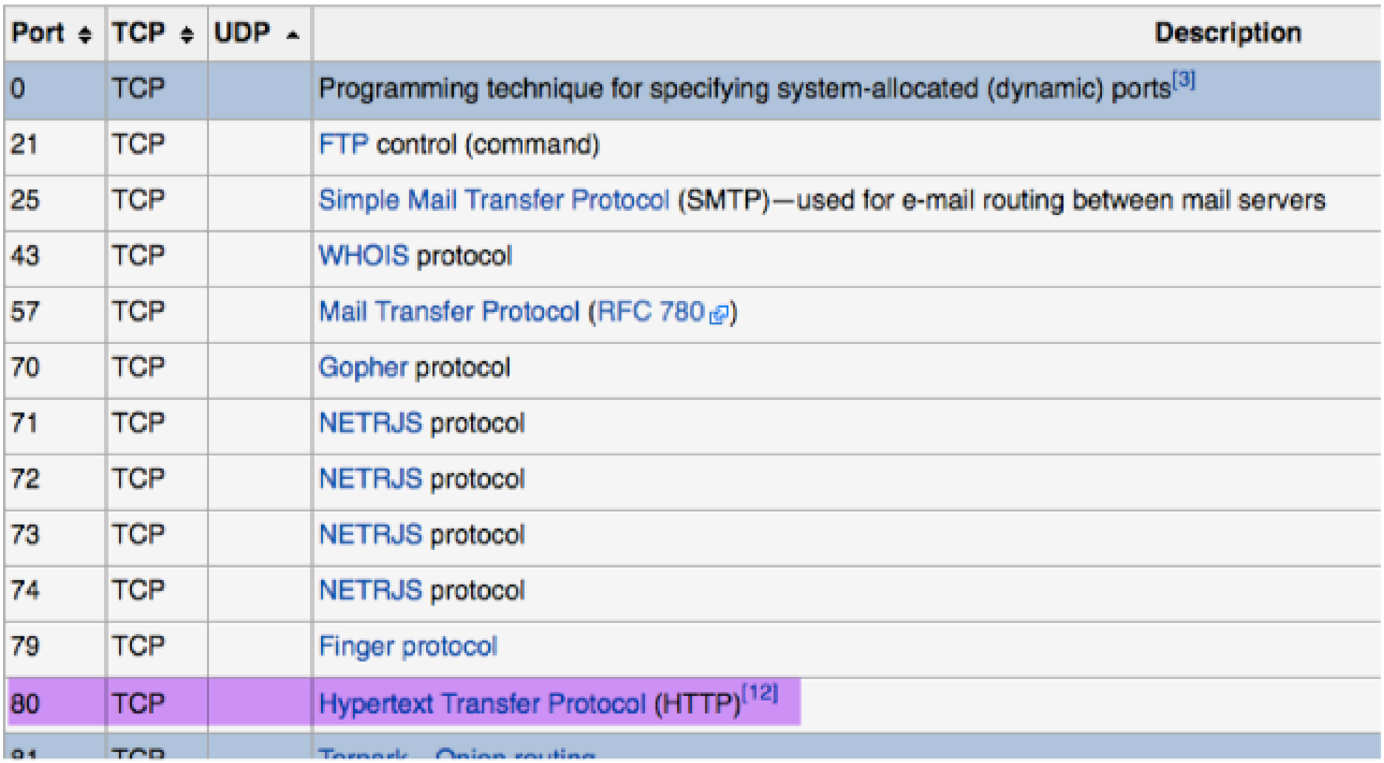
```
sup_tx_mask=0x1
                 src_policy_incomplete_mask=0x1

                 dst_policy_incomplete_mask=0x1

                 class_eq_mask=0x1

                 aclass_mask=0x1ff

                 port_dir_mask=0x1

                 dport_mask=0xffff

                 sport_mask=0xffff

                 tcpflags_mask=0xff

                 ip_opt_mask=0x1

                 ipv6_route_mask=0x1

                 ip_fragment_mask=0x1

                 ip_frag_offset0_mask=0x1

                 ip_frag_offset1_mask=0x1

                 ip_mf_mask=0x1

                 l4_partial_mask=0x1

                 dst_local_mask=0x1

                 routeable_mask=0x1

                 spare_mask=0x7ff

                 v4addr_key_mask=0x1

                 v6addr_key_mask=0x1

                 valid=0x1


module-1#

show platform internal ns table mth_lux_slvz_DHS_SecurityGroupKeyTable0
```

_memif_data 42

```
===============================================================
                      TABLE INSTANCE : 0
===============================================================
ENTRY[000042] =

             sg_label=0x4

             sclass=0x8003

             dclass=0xc003

             prot=0x6

<--



             dport=0x50

<--
```



Decimal	Keyword	Protocol	IPv6 Extension Header	
0	HOPOPT	IPv6 Hop-by-Hop Option	Y	[RFC2460]
1	ICMP	Internet Control Message		[RFC792]
2	IGMP	Internet Group Management		[RFC1112]
3	GGP	Gateway-to-Gateway		[RFC823]
4	IPv4	IPv4 encapsulation		[RFC2003]
5	ST	Stream		[RFC1190][RFC1819]
6	TCP	Transmission Control		[RFC793]
7	CBT	CBT		[Tony_Ballardie]



Port	TCP	UDP	Description
0	TCP		Programming technique for specifying system-allocated (dynamic) ports[3]
21	TCP		FTP control (command)
25	TCP		Simple Mail Transfer Protocol (SMTP)—used for e-mail routing between mail servers
43	TCP		WHOIS protocol
57	TCP		Mail Transfer Protocol (RFC 780)
70	TCP		Gopher protocol
71	TCP		NETRJS protocol
72	TCP		NETRJS protocol
73	TCP		NETRJS protocol
74	TCP		NETRJS protocol
79	TCP		Finger protocol
80	TCP		Hypertext Transfer Protocol (HTTP)[12]
81	TCP		Torpark—Onion routing

```
sup_tx_mask=0x1
               src_policy_incomplete_mask=0x1
               dst_policy_incomplete_mask=0x1
               class_eq_mask=0x1
               aclass_mask=0x1ff
               port_dir_mask=0x1
               sport_mask=0xffff
               tcpflags_mask=0xff
               ip_opt_mask=0x1
               ipv6_route_mask=0x1
               ip_fragment_mask=0x1
               ip_frag_offset0_mask=0x1
               ip_frag_offset1_mask=0x1
               ip_mf_mask=0x1
               l4_partial_mask=0x1
               dst_local_mask=0x1
```

提示：您可以使用相同的方法驗證每個TCAM條目。

排除硬體程式設計問題

本節提供一些有用的故障排除命令和提示。

有用的故障排除命令

遇到問題時，可以使用以下一些有用的命令來查詢枝葉策略管理器錯誤：

<#root>

```
fab1_leaf1#

show system internal policy-mgr event-history errors



1) Event:E_DEBUG, length:84, at 6132 usecs after Mon Sep 8 13:15:56 2014

   [103] policy_mgr_handle_ctx_mrules(779): ERROR: Failed to process prio(1537):
   (null)


2) Event:E_DEBUG, length:141, at 6105 usecs after Mon Sep 8 13:15:56 2014

   [103] policy_mgr_process_mrule_prio_aces(646): ERROR: Failed to insert iptables
   rule for rule(4120) , fentry(5_0) with priority(1537): (null)


[snip]


fab1_leaf1#

show system internal policy-mgr event-histor trace


[1409945922.23737] policy_mgr_ppf_hdl_close_state:562: Got close state callback

[1409945922.23696] policy_mgr_ppf_rdy_ntf_fun:239: StatStoreEnd returned: 0x0(SU

CCESS)

[1409945922.23502] policy_mgr_ppf_rdy_ntf_fun:208: ppf ready notification: sess_

id: (0xFF0104B400005B51)

[1409945922.23475] policy_mgr_ppf_rdy_ntf_fun:205: Got ready notification callba

ck with statustype (4)

[1409945921.983476] policy_mgr_gwrap_handler:992: Dropped...now purging it...

[1409945921.982882] policy_mgr_ppf_goto_state_fun:481: Sess id (0xFF0104B400005B
[snip]

module-1#
show system internal aclqos event-history trace


T [Fri Sep 5 13:18:24.863283] ============= Session End ============

T [Fri Sep 5 13:18:24.862924] Commit phase: Time taken 0.62 ms, usr 0.00 ms,

sys 0.00 ms

T [Fri Sep 5 13:18:24.862302] ppf session [0xff0104b410000087] commit ... npi

nst 1

T [Fri Sep 5 13:18:24.861421] Verify phase: Time taken 0.77 ms, usr 0.00 ms,

sys 0.00 ms

T [Fri Sep 5 13:18:24.860615] ============= Session Begin ============

T [Fri Sep 5 13:18:24.830472] ============= Session End ============

T [Fri Sep 5 13:18:24.830062] Commit phase: Time taken 0.98 ms, usr 0.00 ms,

sys 0.00 ms

T [Fri Sep 5 13:18:24.829085] ppf session [0xff0104b410000086] commit ... npi

nst 1

T [Fri Sep 5 13:18:24.827685] Verify phase: Time taken 2.04 ms, usr 0.00 ms,

sys 0.00 ms

T [Fri Sep 5 13:18:24.825388] ============= Session Begin ============

T [Fri Sep 5 12:32:51.364225] ============= Session End ============

T [Fri Sep 5 12:32:51.363748] Commit phase: Time taken 0.64 ms, usr 0.00 ms,

[snip]
```

提示：有些檔案很大，因此將其傳送到bootflash並在編輯器中檢查它們比較容易。

<#root>

```
module-1#
show system internal aclqos ?
```

```
asic            Asic information
brcm            Broadcam information
database        Database
event-history   Show various event logs of ACLQOS
mem-stats       Show memory allocation statistics of ACLQOS
prefix          External EPG prefixes
qos             QoS related information
range-resource  Zoning rules L4 destination port range resources
regions         Security TCAM priority regions
span            SPAN related information
zoning-rules    Show zoning rules

module-1#
show system internal aclqos event-history ?


errors        Show error logs of ACLQOS


msgs          Show various message logs of ACLQOS
ppf           Show ppf logs of ACLQOS
ppf-parse     Show ppf-parse logs of ACLQOS
prefix        Show prefix logs of ACLQOS
qos           Show qos logs of ACLQOS
qos-detail    Show detailed qos logs of ACLQOS
span          Show span logs of ACLQOS
span-detail   Show detailed span logs of ACLQOS
trace         Show trace logs of ACLQOS


trace-detail  Show detailed trace logs of ACLQOS
zoning-rules  Show detailed logs of ACLQOS
```

疑難排解提示

以下是一些有用的故障排除提示：

- 如果您似乎遇到TCAM耗盡問題，請檢查UI或CLI中是否存在與問題規則相關的故障。可以報告此故障：

  ```
  <#root>

  Fault F1203 - Rule failed due to hardware programming error.
  ```

  在專用積體電路(ASIC)中，一個規則可以採用多個TCAM條目。要檢視ASIC上的條目數，請輸入以下命令：

  ```
  <#root>

  fab1-leaf1#

  vsh_lc



  module-1#

  show platform internal ns table-health


  VLAN STATE curr usage: 0 - size: 4096
  QQ curr usage: 0 - size: 16384
  SEG STATE curr usage: 0 - size: 4096
  SRC TEP curr usage: 0 - size: 4096
  POLICY KEY curr usage: 0 - size: 1
  SRC VP curr usage: 0 - size: 4096


  SEC GRP curr usage: 43 - size: 4096
  ```

  註：在本例中，有43個條目。此使用情況也會報告給eqptCapacity類中的APIC。

- 當存在多個匹配項時，TCAM查詢將返回較低的hw-index。若要驗證索引，請輸入以下命令：

  ```
  <#root>

  show system internal aclqos zoning-rule
  ```

  進行故障排除時，您可以觀察由any-any-implicit規則引起的丟棄。此規則始終位於底部，這表示由於規則不存在，資料包被丟棄。這可能是由於配置錯誤造成的，或者策略元素管理器沒有按預期對其進行程式設計。

- pcTags可以具有本地或全域性範圍：

系統保留pcTag — 此pcTag用於系統內部規則(1-15)。

全域性範圍的pcTag — 此pcTag用於共用服務(16-16385)。

本地作用域的pcTag — 每個VRF在本地使用此pcTag(範圍為16386-65535)。

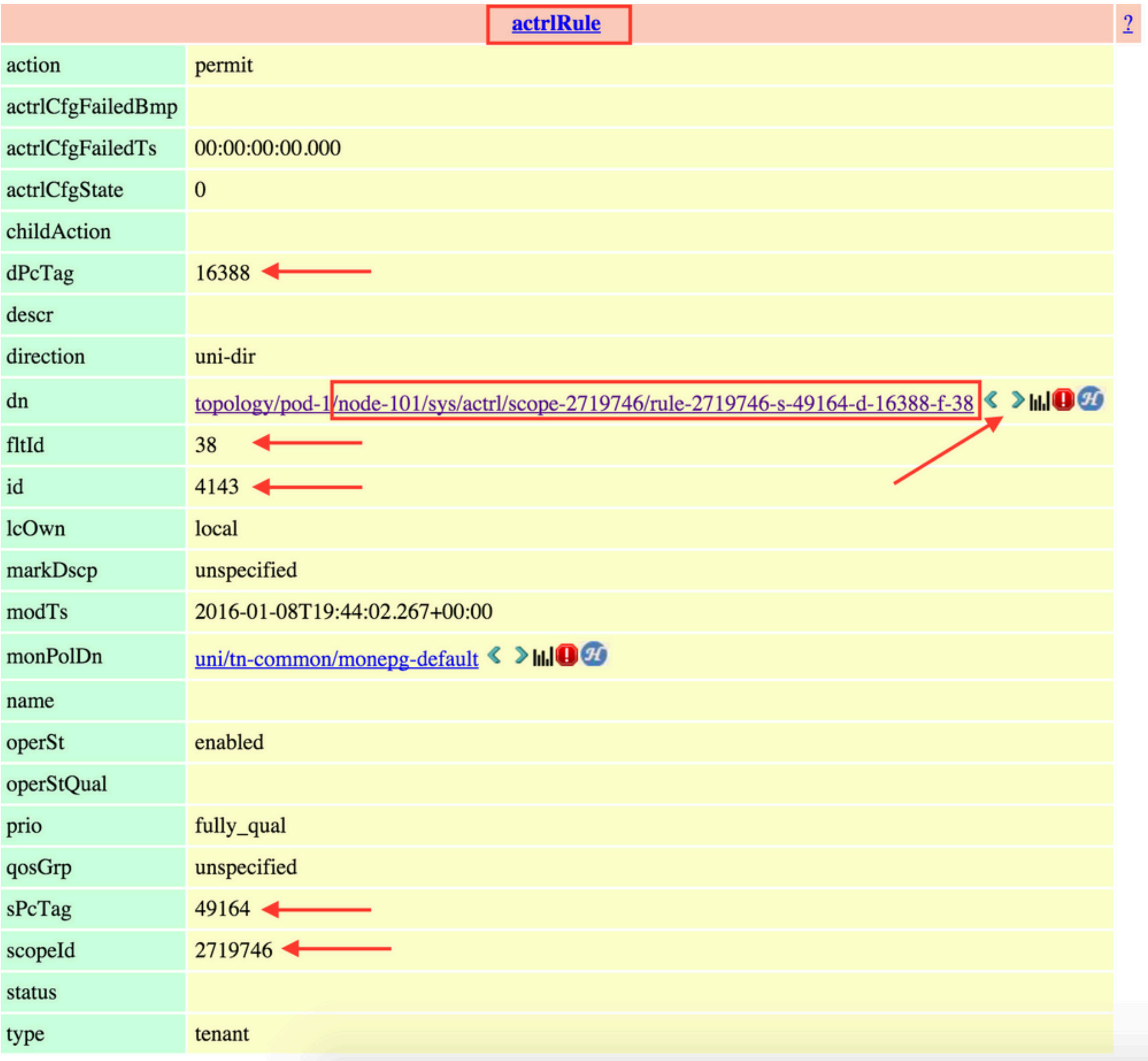
進行故障排除時，快速檢視值的長度即可指示其範圍。

從規則ID派生合約名稱

通常，當進行故障排除時，工程師會檢視分割槽規則。在某些情況下，EPG/pcTag具有許多合約，因此進行故障排除可能非常麻煩。本節將通過交換機CLI上顯示的規則ID來確定EPG/pcTags之間正在使用的合約名稱。

若要開始，請查詢具體的合約/規則對象actrlRule（如果需要），按屬性id value: rule-d縮小搜尋範圍

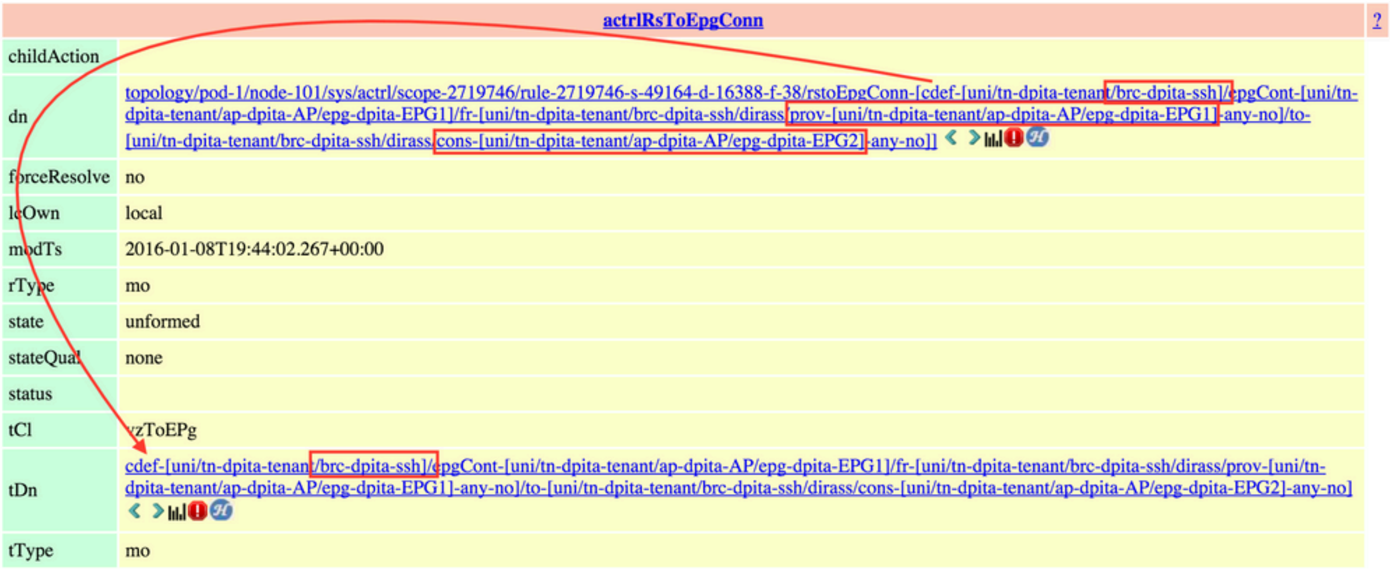
找到正確的規則後，按一下DN上的綠色箭頭檢視actrlRule對象的子項。我們的答案在孩子身上。



actrlRule ?

action	permit
actrlCfgFailedBmp	
actrlCfgFailedTs	00:00:00:00.000
actrlCfgState	0
childAction	
dPcTag	16388
descr	
direction	uni-dir
dn	topology/pod-1/node-101/sys/actrl/scope-2719746/rule-2719746-s-49164-d-16388-f-38
fltId	38
id	4143
lcOwn	local
markDscp	unspecified
modTs	2016-01-08T19:44:02.267+00:00
monPolDn	uni/tn-common/monepg-default
name	
operSt	enabled
operStQual	
prio	fully_qual
qosGrp	unspecified
sPcTag	49164
scopeId	2719746
status	
type	tenant

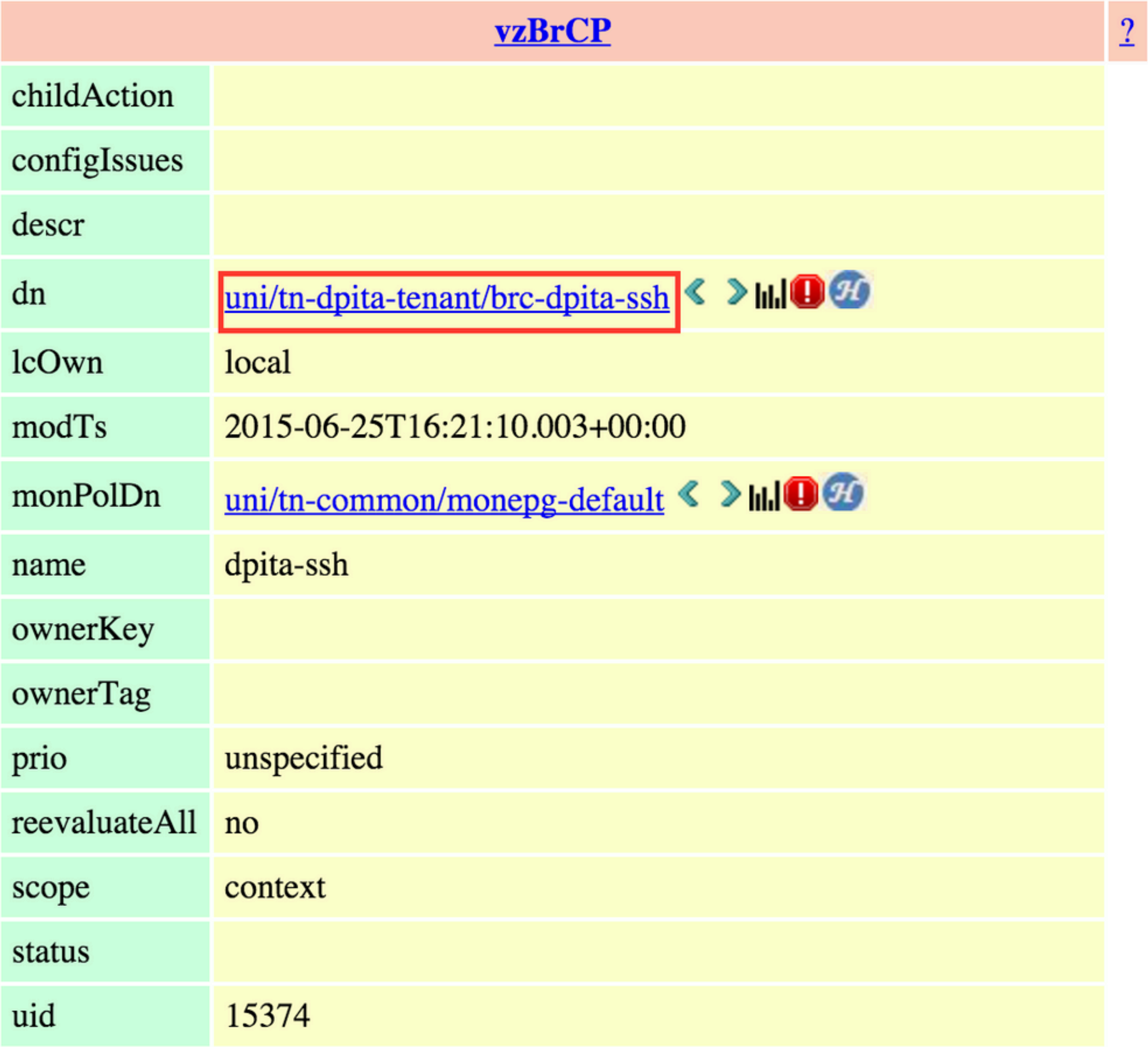
這裡的子對象是actrlRsToEpgConn。通常，可以有兩種，每個EPG一個。此對象的DN顯示應用合約的兩個EPG以及方向（提供商或消費者），最重要的是，顯示合約對象名稱。



actrlRsToEpgConn	?
childAction	
dn	topology/pod-1/node-101/sys/actrl/scope-2719746/rule-2719746-s-49164-d-16388-f-38/rstoEpgConn-[cdef-[uni/tn-dpita-tenant/brc-dpita-ssh]/epgCont-[uni/tn-dpita-tenant/ap-dpita-AP/epg-dpita-EPG1]/fr-[uni/tn-dpita-tenant/brc-dpita-ssh/dirass/prov-[uni/tn-dpita-tenant/ap-dpita-AP/epg-dpita-EPG1]-any-no]/to-[uni/tn-dpita-tenant/brc-dpita-ssh/dirass/cons-[uni/tn-dpita-tenant/ap-dpita-AP/epg-dpita-EPG2]-any-no]]
forceResolve	no
lcOwn	local
modTs	2016-01-08T19:44:02.267+00:00
rType	mo
state	unformed
stateQual	none
status	
tCl	vzToEPg
tDn	cdef-[uni/tn-dpita-tenant/brc-dpita-ssh]/epgCont-[uni/tn-dpita-tenant/ap-dpita-AP/epg-dpita-EPG1]/fr-[uni/tn-dpita-tenant/brc-dpita-ssh/dirass/prov-[uni/tn-dpita-tenant/ap-dpita-AP/epg-dpita-EPG1]-any-no]/to-[uni/tn-dpita-tenant/brc-dpita-ssh/dirass/cons-[uni/tn-dpita-tenant/ap-dpita-AP/epg-dpita-EPG2]-any-no]
tType	mo

如突出顯示，此案例中的合約名稱為brc-dpita-ssh。

如果需要，請查詢vzBrCP以查詢正確的合約。



vzBrCP		?
childAction		
configIssues		
descr		
dn	uni/tn-dpita-tenant/brc-dpita-ssh	
lcOwn	local	
modTs	2015-06-25T16:21:10.003+00:00	
monPolDn	uni/tn-common/monepg-default	
name	dpita-ssh	
ownerKey		
ownerTag		
prio	unspecified	
reevaluateAll	no	
scope	context	
status		
uid	15374	

關於此翻譯