

# Verifica di contratti e regole in ACI Fabric

## Sommario

---

[Introduzione](#)

[Topologia](#)

[Panoramica del processo](#)

[Identifica la regola di contratto/suddivisione in zone utilizzata](#)

[Verifica della programmazione hardware](#)

[Risoluzione dei problemi di programmazione hardware](#)

[Comandi utili per la risoluzione dei problemi](#)

[Suggerimenti per la risoluzione dei problemi](#)

[Deriva nome contratto da ID regola](#)

---

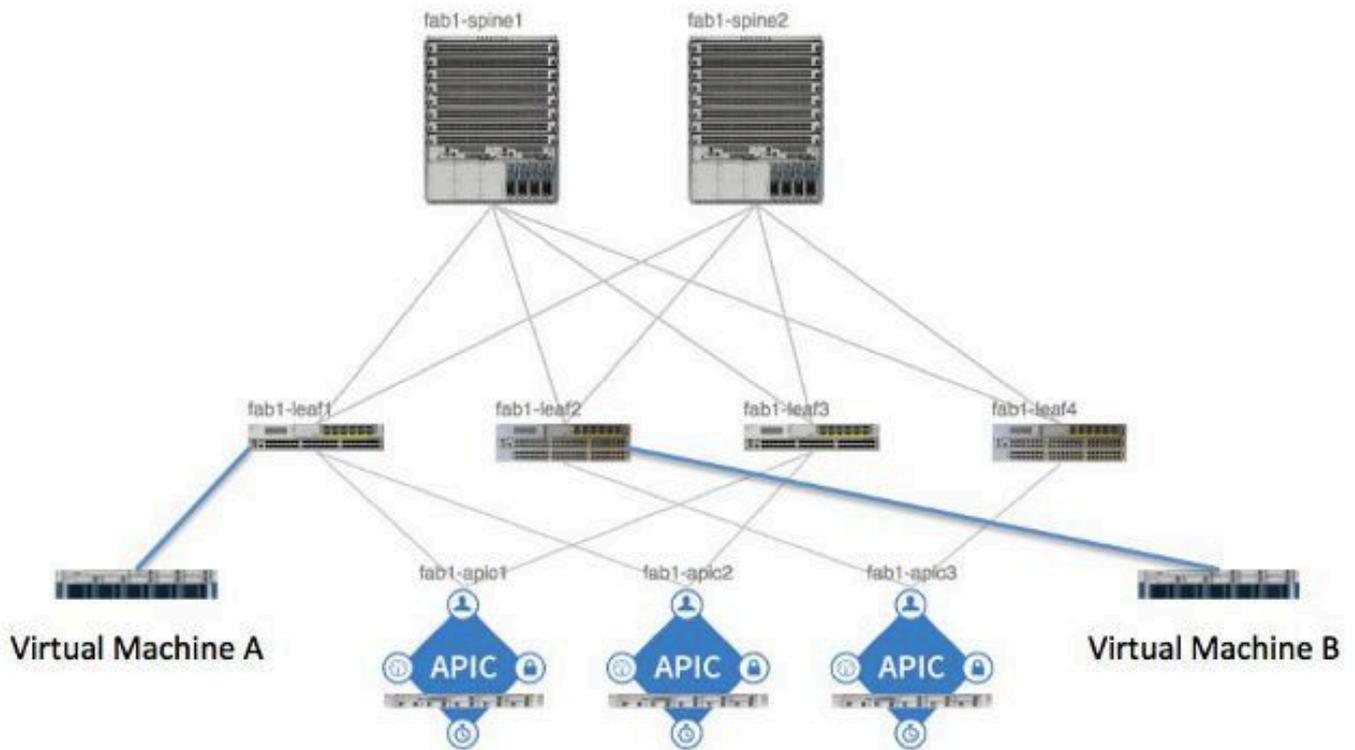
## Introduzione

In questo documento viene descritto come verificare che i contratti siano configurati e funzionino correttamente nell'infrastruttura ACI (Application Centric Infrastructure).

## Topologia

Nell'esempio utilizzato in questo documento, Virtual Machine-A (VM) è collegata a Leaf1 ed è presente un contratto che consente la comunicazione con VM-B, collegato a Leaf2. Il contratto consente sia l'uso del protocollo ICMP (Internet Control Message Protocol) che del protocollo HTTP.

Nell'immagine viene illustrata la topologia:



## Panoramica del processo

Si tratta dell'interazione e del flusso delle politiche per i contratti e le regole:

1. Policy Manager su Application Policy Infrastructure Controller (APIC) comunica con Policy Element Manager sullo switch.
2. Policy Element Manager sullo switch programma Object Store sullo switch.
3. Il Policy Manager sullo switch comunica con il client Access Control List Quality of Service (ACLQOS) sullo switch.
4. Il client ACLQOS programma l'hardware.

## Identifica la regola di contratto/suddivisione in zone utilizzata

Di seguito è riportato un esempio di output del comando `show zoning-rule` restituito dalla foglia prima di aggiungere il contratto per i due gruppi di endpoint (EPG).

```
<#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]

Questo è lo stesso output del comando dopo l'aggiunta del contratto in modo che i due EPG possano comunicare tra loro:

```
<#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]

 Nota: si notino i nuovi ID regola (4131 e 4132) aggiunti, gli ID filtro 7 e 6 e l'ambito 2686976.

 Attenzione: questo output del comando consente di individuare facilmente le regole da esaminare in un sistema lab. Tuttavia, le modifiche dinamiche che si verificano possono risultare molto complesse in un ambiente di produzione.

Un altro metodo che è possibile utilizzare per individuare le regole di interesse consiste nell'utilizzare Visore. Eseguire una ricerca di fvCtx nell'oggetto gestito del contesto (MO). È quindi possibile cercare il proprio nome distinto (DN) di contesto specifico, come illustrato di seguito:

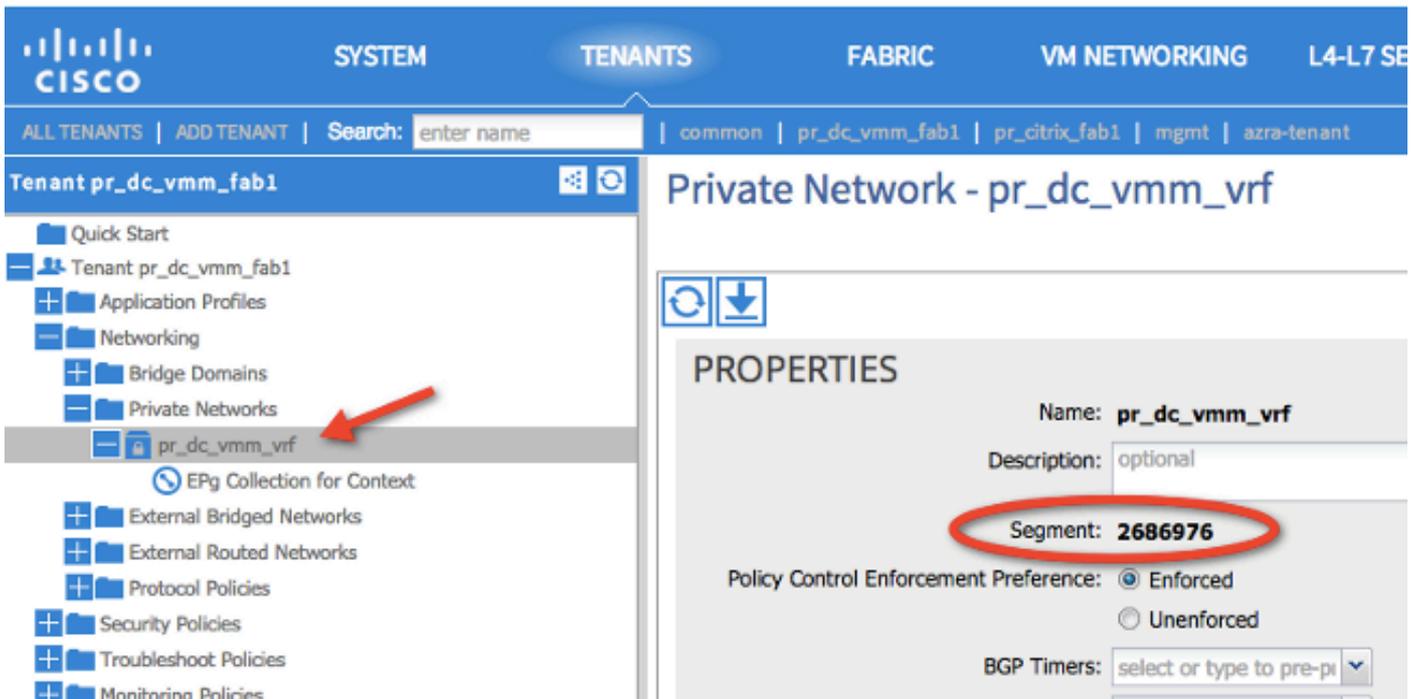


The screenshot shows the APIC Object Store Browser interface. At the top, there is a search bar with the text 'fvCtx' entered. Below the search bar, there are fields for 'Property', 'Op', 'Val1', and 'Val2'. A 'Run Query' button is visible. Below the search bar, there are links for 'Display URL of last query' and 'Display last response'. The search results are displayed in a table with the following columns: 'childAction', 'descr', and 'dn'. The first row is highlighted in orange and contains the value 'fvCtx' in the 'childAction' column. The 'dn' column shows the value 'uni/te-infra/ctx-overlay-1'.

Prendere nota della portata di tale contesto. È possibile utilizzare questa opzione per eseguire il mapping all'output del comando show-zoning-rule in modo da individuare le regole su cui eseguire le query:

fvCtx		1
childAction		
descr		
dn	<a href="#">uni/tn-pr_dc_vmm_fab1/ctx-pr_dc_vmm_vrf</a> < >	
knwMeastAct	permit	
lcOwn	local	
modTs	2014-09-03T09:32:36.625-04:00	
monPolDn	<a href="#">uni/tn-common/monepg-default</a> < >	
name	pr_dc_vmm_vrf	
ownerKey		
ownerTag		
pcEntPref	enforced	
pcTag	32770	
scope	2686976	
seg	2686976	
status		
uid	15374	

È inoltre possibile identificare l'ID/ambito del segmento per il contesto dall'interfaccia utente (UI), come illustrato di seguito:



Questo ambito corrisponde a quello mostrato nell'output del comando show zoning-rules:

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

Dopo aver ottenuto le informazioni sull'ID di ambito e aver identificato gli ID della regola e del filtro, è possibile utilizzare il comando successivo per verificare di aver applicato i nuovi filtri (e non i messaggi di rifiuto impliciti tra gli EPG). Il messaggio di rifiuto implicito è incluso in modo che, per impostazione predefinita, gli EPG non possano comunicare.

In questo output del comando Leaf1, Filter-6 (f-6) è in aumento:

```
<#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

In questo output del comando Leaf2, Filter-7 (f-7) è in aumento:

<#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



Suggerimento: la conoscenza dell'ambito, dell'ID regola, della destinazione, dei tag pc di origine e del filtro è importante per tentare di risolvere ulteriormente questo problema. È inoltre utile conoscere gli EPG tra i quali esiste l'ID regola.

---

È possibile eseguire una ricerca sul disco magneto-ottico con il nome DN fvAEPg e grep per il tag pcTag specifico tramite il comando moquery, come mostrato di seguito:

```
<#root>
```

```
admin@RTP_Apic1:~>
```

```
moquery -c fvAEPg | grep 49155 -B 5
```

```
dn : uni/tn-Prod/ap-commercespace/
```

```
epg-Web
```

```
lcOwn : local
```

```
matchT : AtleastOne
```

```
modTs : 2014-10-16T01:27:35.355-04:00
```

```
monPolDn : uni/tn-common/monepg-default
```

```
pcTag : 49155
```

È inoltre possibile utilizzare l'opzione filter con il comando moquery, come illustrato di seguito:

```
<#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

## Verifica della programmazione hardware

È ora possibile verificare la voce hardware per la regola. Per visualizzare le informazioni sull'hardware, immettere il comando `show platform internal ns table mth_lux_slvz_DHS_SecurityGroupStatTable_memif_data` in entrata (comando `vsh_lc`):

```
module-1# show platform internal ns table mth_lux_slvz_DHS_SecurityGroupStatTable_memif_data ingress
error opening file
: No such file or direct
Last login: Fri Sep  5 1
=====
[Restored]
TABLE INSTANCE : 0
=====
ENTRY[000010] = pkt_cnt=0x5176e
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
ENTRY[000029] = pkt_cnt=0x5d7e2
ENTRY[000041] = pkt_cnt=0x6360
ENTRY[000050] = pkt_cnt=0x2a05
ENTRY[000052] = pkt_cnt=0x5ec
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
=====
TABLE INSTANCE : 0
=====
ENTRY[000010] = pkt_cnt=0x517cf
ENTRY[000011] = pkt_cnt=0x7d9f
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
ENTRY[000041] = pkt_cnt=0x64e0
ENTRY[000050] = pkt_cnt=0x2a05
ENTRY[000052] = pkt_cnt=0x5ec
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
```

In questo esempio, la voce hardware 41 (ENTRY [000041]) viene incrementata.

 Nota: il comando precedente illustrato è utilizzato per l'ASIC Northstar. Il comando utilizzato per Donner o Donner+ è show platform internal ns table mth\_luxh\_slvy\_DHS\_SecurityGroupStatTable\_memif\_data.

 Nota: l'utilizzo di questo comando non è pratico in un ambiente di produzione, ma è possibile

 utilizzare gli altri comandi descritti in questa sezione.

Ricordare la regola (4132) e l'ambito (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

Immettere questo comando per determinare l'ID regola per il mapping delle voci dell'indice hardware TCAM (Ternary Content-Addressable Memory) e filtrare in base all'ID regola e/o all'ID filtro:

```
<#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
```

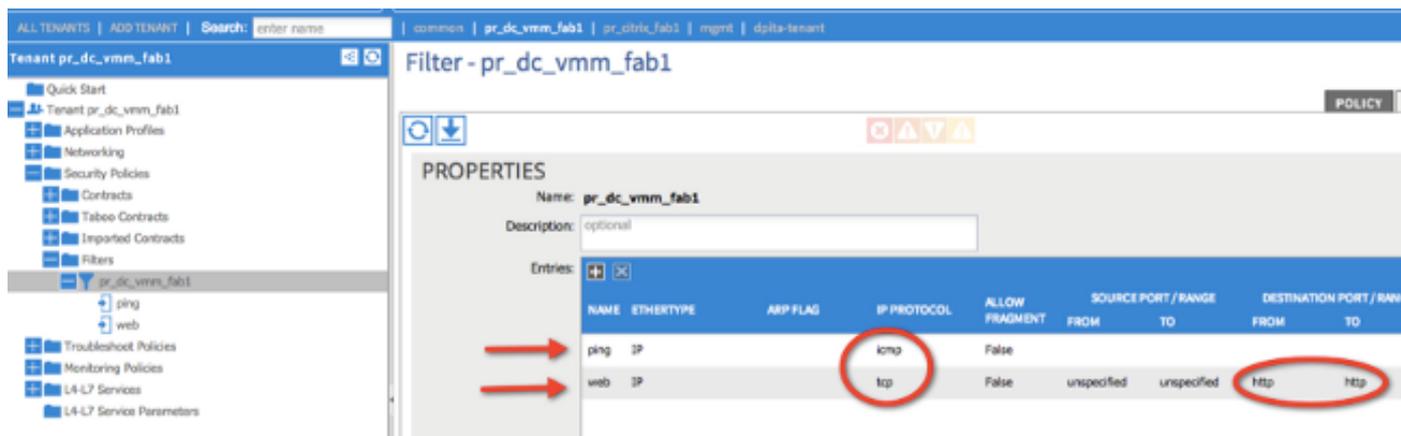
```
=== Region priority: 771 (rule prio: 3 entry: 3)===  
sw_index = 67 |
```

```
hw_index = 42
```

[snip]

Per questo esempio, la combinazione di interessi EPG di origine e destinazione è 32771=0x8003, 49155=0xC003. Pertanto, è possibile considerare tutte le voci TCAM per queste classi di origine e di destinazione che corrispondono agli ID regola (4131 e 4132) e agli ID filtro (6 e 7).

In questo esempio, alcune di queste voci TCAM vengono scaricate. Di seguito è riportata la configurazione del contratto che consente i ping e il traffico Web per questi EPG:



<#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)
```

 Nota: il comando precedente illustrato è utilizzato per l'ASIC Northstar. Il comando utilizzato per Donner o Donner+ è show platform internal ns table

 mth\_luxh\_slvq\_DHS\_SecurityGroupKeyTable0\_memif\_data.

Decimal <input type="checkbox"/>	Keyword <input type="checkbox"/>	Protocol <input type="checkbox"/>	IPv6 Extension Header <input type="checkbox"/>	
0	HOPOPT	IPv6 Hop-by-Hop Option	Y	<a href="#">[RFC2460]</a>
1	ICMP	Internet Control Message		<a href="#">[RFC792]</a>
2	IGMP	Internet Group Management		<a href="#">[RFC1112]</a>

<#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
  t4_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	<a href="#">[RFC2460]</a>
1	ICMP	Internet Control Message		<a href="#">[RFC792]</a>
2	IGMP	Internet Group Management		<a href="#">[RFC1112]</a>
3	GGP	Gateway-to-Gateway		<a href="#">[RFC823]</a>
4	IPv4	IPv4 encapsulation		<a href="#">[RFC2003]</a>
5	ST	Stream		<a href="#">[RFC1190]</a> <a href="#">[RFC1819]</a>
6	TCP	Transmission Control		<a href="#">[RFC793]</a>
7	CBT	CBT		<a href="#">[Tony Ballardie]</a>

Port ↕	TCP ↕	UDP ▲	Description
0	TCP		Programming technique for specifying system-allocated (dynamic) ports <sup>[3]</sup>
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 <a href="#">↗</a> )
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) <sup>[12]</sup>
81	TCP		Terminology: 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

---

 Suggerimento: è possibile verificare ciascuna voce TCAM con lo stesso metodo.

---

## Risoluzione dei problemi di programmazione hardware

In questa sezione vengono forniti alcuni utili comandi e suggerimenti per la risoluzione dei problemi.

### Comandi utili per la risoluzione dei problemi

Di seguito sono riportati alcuni comandi utili che è possibile utilizzare per individuare gli errori di Gestione criteri foglia quando si verificano dei problemi:

```
<#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_mrle_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]



Suggerimento: alcuni file sono di grandi dimensioni, quindi è più facile inviarli al bootflash ed esaminarli in un editor.

---

<#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

## Suggerimenti per la risoluzione dei problemi

Di seguito sono riportati alcuni suggerimenti utili per la risoluzione dei problemi:

- Se sembra che si sia verificato un problema di esaurimento TCAM, controllare l'interfaccia utente o la CLI per individuare eventuali errori associati alla regola in questione. Questo errore può essere segnalato:

```
<#root>
```

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

Una regola può richiedere più voci TCAM nel circuito integrato specifico dell'applicazione (ASIC, Application-Specific Integrated Circuit). Per visualizzare il numero di voci nell'ASIC, immettere i seguenti comandi:

```
<#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
```



Nota: nell'esempio sono presenti 43 voci. Questo utilizzo viene segnalato anche all'APIC nella classe eqptCapacity.

---

- In presenza di più corrispondenze, la ricerca TCAM restituisce l'indice hw inferiore. Per verificare l'indice, immettere questo comando:

```
<#root>
```

```
show system internal aclqos zoning-rule
```

Durante la risoluzione dei problemi, è possibile osservare il calo causato dalla regola any-any-implicit. Questa regola si trova sempre in basso, il che significa che il pacchetto viene scartato perché non esiste una regola. Ciò è dovuto a una configurazione errata oppure Policy Element Manager non la programma come previsto.

- I tag pc possono avere un ambito locale o globale:

PcTag riservato del sistema - Questo pcTag viene utilizzato per le regole interne del sistema (1-15).

PcTag con ambito globale - Questo pcTag viene utilizzato per il servizio condiviso (16-16385).

PcTag con ambito locale - Questo pcTag viene utilizzato localmente per VRF (intervallo da 16386-65535).

Quando si esegue la risoluzione dei problemi, la lunghezza del valore ne indica l'ambito.

## Deriva nome contratto da ID regola

Spesso, in un caso di risoluzione dei problemi, un tecnico sta prendendo in considerazione le regole di zoning. In alcuni casi, un EPG/pcTag ha molti contratti e può essere difficile risolvere il problema. In questa sezione viene spiegato in che modo determinare il nome del contratto in uso tra EPG/pcTags in base all'ID della regola visualizzato nella CLI dello switch.

Per iniziare, eseguire una query per l'oggetto contratto/regola concreto `actrlRule`, se lo si desidera, restringere la ricerca in base alla proprietà: `id valore: rule-d`

Una volta trovata la regola corretta, fare clic sulla freccia verde sul DN per visualizzare gli oggetti figlio di `actrlRule`. I bambini sono la nostra risposta.

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

L'oggetto figlio è actrlRsToEpgConn. Di solito ce ne possono essere due, una per ogni EPG. Il DN dell'oggetto mostra i due EPG tra cui viene applicato il contratto, oltre alla direzione (fornitore o consumatore) e, cosa più importante, il nome dell'oggetto del contratto.

actrlRsToEpgConn	
childAction	
dn	<a href="#">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]]</a> < >   
forceResolve	no
leOwn	local
modTs	2016-01-08T19:44:02.267+00:00
rType	mo
state	unformed
stateQual	none
status	
tCl	vzToEPg
tDn	<a href="#">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]]</a> < >   
tType	mo

Come evidenziato, in questo caso il nome del contratto è brc-dpita-ssh.

Se necessario, cercare vzBrCP per trovare il contratto corretto.

## vzBrCP

?

childAction	
configIssues	
descr	
dn	<a href="#">uni/tn-dpita-tenant/brc-dpita-ssh</a> < >     ! H
lcOwn	local
modTs	2015-06-25T16:21:10.003+00:00
monPolDn	<a href="#">uni/tn-common/monepg-default</a> < >     ! H
name	dpita-ssh
ownerKey	
ownerTag	
prio	unspecified
reevaluateAll	no
scope	context
status	
uid	15374

## Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).