Gedrag van ACL in PBR op Nexus 7K dat zowel L3- als L4-informatie bevat

Inhoud

Inleiding

Dit document beschrijft het gedrag van Policy-Based Routing (PBR) op Nexus-switches wanneer u filter op basis van Layer 3 (L3) en Layer 4 (L4) informatie.

Achtergrondinformatie

Als u een reeks in PBR toevoegt om specifieke L4 informatie aan te passen, aangezien een functie N7K waarden voor Access Control Entry (ACE's) creëert en een fragment ACE automatisch wordt gemaakt dat overeenkomt met de L3 informatie die in de matchreeks is gespecificeerd. In geval van gefragmenteerde pakketten bevat het eerste pakket dat als eerste fragment bekend is de L4-header en wordt correct aangepast in de ACL-toegangscontrolelijst (ACL). De volgende fragmenten die bekend staan als niet-initiële fragmenten bevatten echter geen L4-informatie en dus als het L3-gedeelte van de ACL-entry-overeenkomsten het niet-initiële fragment is toegestaan. Er moet dus uiterste zorgvuldigheid worden betracht bij het filteren van het verkeer op basis van L4-informatie, aangezien de niet-initiële fragmenten bij gebrek aan L4-informatie onjuist kunnen worden gerouteerd.

Topologie

De LAN-router is op interface E2.1 en VLAN 700 met Nexus verbonden. Het is vereist om het verkeer dat overeenkomt met Simple Network Management Protocol (SNMP), Web enz., te richten op Optimaliser en al het andere verkeer direct om E2/2 in de richting van firewall te zetten. PBR is ingesteld op Switch Virtual Interface (SVI) VLAN700 op Nexus-apparaat. Configuratie voor het zelfde wordt hier verstrekt. Volgorde 70 in de route-kaart voorwaarts al ander verkeer naar Firewall. Er is een nieuwe vereiste dat al het verkeer met UDP-poort 920x via Optimizer moet verlopen, want deze reeks 50 wordt toegevoegd in de route-kaart.

Zie hier hoe PBR reageert op gefragmenteerde en niet-gefragmenteerde pakketten die in volgorde 50 zijn ingedrukt en zowel L3 als L4 informatie matchen.

Hier is de configuratie op Nexus interface VLAN700 om het verkeer dat op E2/1 komt om te leiden:

```
interface Vlan700

 no shutdown

 mtu 9000

 vrf member ABC

 no ip redirects

 ip address 10.11.25.25/28

 ip policy route-map In_to_Out


Nexus# show route-map In_to_Out

route-map In_to_Out, permit, sequence 3

 Match clauses:

   ip address (access-lists): Toolbar

 Set clauses:

   ip next-hop 10.3.22.13

route-map In_to_Out, permit, sequence 5

 Match clauses:

   ip address (access-lists): Internet

 Set clauses:

   ip next-hop 10.11.25.19

route-map In_to_Out, permit, sequence 7

 Match clauses:

   ip address (access-lists): Web

 Set clauses:

   ip next-hop 10.11.25.19

route-map In_to_Out, permit, sequence 10

 Match clauses:

   ip address (access-lists): In_to_Out_Internet

 Set clauses:
```

```
   ip next-hop 10.11.25.23
route-map In_to_Out, permit, sequence 30
 Match clauses:
   ip address (access-lists): In_to_Out_www
 Set clauses:
   ip next-hop 10.11.25.23
route-map In_to_Out, permit, sequence 35
 Match clauses:
   ip address (access-lists): In_to_Out_https
 Set clauses:
   ip next-hop 10.11.25.23
route-map In_to_Out, permit, sequence 40
 Match clauses:
   ip address (access-lists): In_to_Out_8080
 Set clauses:
   ip next-hop 10.11.25.23
route-map In_to_Out, permit, sequence 50
 Match clauses:
   ip address (access-lists): UDP_Traffic
 Set clauses:
   ip next-hop 10.11.25.23 >>>>>>>>>>>>>>>>>>> Towards Optimizer
route-map In_to_Out, permit, sequence 70
 Match clauses:
   ip address (access-lists): To_Firewall
 Set clauses:
    ip next-hop . 10.22.45.63 >>>>>>>>>>>>>>>> Towards Firewall


Nexus# show ip access-lists UDP_Traffic
IP access list UDP_Traffic
10 permit udp any any eq 9201
20 permit udp any any eq 9202
30 permit udp any any eq 9203
```

```
Nexus# sh ip access-lists To_Firewall

IP access list To_Firewall

       10 permit ip any any
```

Zodra de op beleid gebaseerde routing op SVI is ingesteld, maakt Nexus een ingang in hardware voor hetzelfde product. Laten we nu kijken naar de hardwareprogrammering voor de PBR op module 2 van Nexus:

```
Nexus# show system internal access-list vlan 700 input entries detail module 2

Flags: F - Fragment entry  E - Port Expansion

      D - DSCP Expansion  M - ACL Expansion

      T - Cross Feature Merge Expansion

INSTANCE 0x0

---------------

 Tcam 1 resource usage:

 ----------------------

 Label_b = 0x201

  Bank 0

  ------

    IPv4 Class

      Policies: PBR(GGSN_Toolbar)

      Netflow profile: 0

      Netflow deny profile: 0

      Entries:

        [Index] Entry [Stats]

        ---------------------

 [0019:000f:000f] prec 1 permit-routed ip 0.0.0.0/0 224.0.0.0/4   [0]

 [002d:0024:0024] prec 1 redirect(0x5d)-routed tcp 1.1.22.80/28 0.0.0.0/0 eq 80 flow-label 80
[0]

 [002e:0025:0025] prec 1 redirect(0x5d)-routed tcp 1.1.22.80/28 0.0.0.0/0 fragment   [0]

 [002f:0026:0026] prec 1 redirect(0x5d)-routed tcp 1.1.22.80/28 0.0.0.0/0 eq 8080 flow-label
8080  [0]

 [0030:0027:0027] prec 1 redirect(0x5d)-routed tcp 1.1.22.80/28 0.0.0.0/0 fragment   [0]

 [0031:0028:0028] prec 1 redirect(0x5d)-routed tcp 1.1.22.48/28 0.0.0.0/0 eq 80 flow-label 80
[0]
```

```
 [0032:0029:0029] prec 1 redirect(0x5d)-routed tcp 1.1.22.48/28 0.0.0.0/0 fragment   [0]

 [0033:002a:002a] prec 1 redirect(0x5d)-routed tcp 1.1.22.48/28 0.0.0.0/0 eq 8080 flow-label
8080  [0]

 [0034:002b:002b] prec 1 redirect(0x5d)-routed tcp 1.1.22.48/28 0.0.0.0/0 fragment   [0]

 [0035:002c:002c] prec 1 permit-routed ip 1.1.22.24/29 0.0.0.0/0   [0]

 [0036:002d:002d] prec 1 permit-routed ip 1.1.22.32/28 0.0.0.0/0   [0]

 [0037:002e:002e] prec 1 permit-routed ip 1.1.22.64/28 0.0.0.0/0   [0]

 [0038:002f:002f] prec 1 permit-routed ip 1.1.22.80/28 0.0.0.0/0   [0]

 [003d:0033:0033] prec 1 permit-routed ip 1.1.22.96/28 0.0.0.0/0   [0]

 [003e:0034:0034] prec 1 permit-routed tcp 0.0.0.0/0 196.11.146.149/32 eq 25 flow-label 25  [0]

 [0059:004f:004f] prec 1 permit-routed tcp 0.0.0.0/0 196.11.146.149/32 fragment   [0]

 [005a:0050:0050] prec 1 redirect(0x5e)-routed ip 1.1.22.16/29 0.0.0.0/0   [0]

 [005b:0051:0051] prec 1 redirect(0x5e)-routed tcp 0.0.0.0/0 0.0.0.0/0 eq 80 flow-label 80  [0]

 [005c:0052:0052] prec 1 redirect(0x5e)-routed tcp 0.0.0.0/0 0.0.0.0/0 fragment   [0]

 [005d:0053:0053] prec 1 redirect(0x5e)-routed tcp 0.0.0.0/0 0.0.0.0/0 eq 443 flow-label 443
[0]

 [005e:0054:0054] prec 1 redirect(0x5e)-routed tcp 0.0.0.0/0 0.0.0.0/0 fragment   [0]

 [005f:0055:0055] prec 1 redirect(0x5e)-routed tcp 0.0.0.0/0 0.0.0.0/0 eq 8080 flow-label 8080
[0]

 [0060:0056:0056] prec 1 redirect(0x5e)-routed tcp 0.0.0.0/0 0.0.0.0/0 fragment   [0]

********************Sequence 50 is to match the traffic for UDP ports
9201/9202/9203**********************

 [0061:0057:0057] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9201 flow-label 9201
[0]

 [0062:0058:0058] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment   [0]

 [0063:0059:0059] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9202 flow-label 9202
[0]

 [0064:005a:005a] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment   [0]

 [0065:005b:005b] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9203 flow-label 9203
[0]

 [0066:005c:005c] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment   [0]

********************Sequence 70 is to send all other traffic to Firewall**********************

 [0067:005d:005d] prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0   [23]

 [0068:005e:005e] prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0   [0]
```

U ziet dat er naast een toegangslijst die **overeenkomt met** het **udp 0.0.0.0/0 0.0.0.0/0 eq 9201,** nog een artikel is dat overeenkomt met het **fragment 0.0.0.0/0 0.0.0.0/0 van** de **fragmenten**, maar dat artikel geen UDP-poortinformatie heeft. Deze ingang is gelijkwaardig aan elke andere die het

pakket UDP aanpast, zodat de pakketten voor andere UDP havens ook worden aangepast in deze opeenvolging die door hardware wordt gegenereerd.

Testcase 1: Verkeer geïnitieerd van LAN-router naar firewall

- Het pakket dat de Nexus bereikt was niet-gefragmenteerd en dus kwam het verkeer overeen zoals verwacht in PBR.
- Hij is op de juiste manier naar de firewall gericht en kan worden gezien in firewalls die op firewall werken.

UDP packet -port 500

```
*Mar 26 04:07:48.959: IP: s=1.1.1.1 (GigabitEthernet0/0), d=3.3.3.3, len 28, rcvd 4  -à
Traffic entering from Nexus interface

*Mar 26 04:07:48.959:     UDP src=500, dst=500
```

TCP packet - port 80

```
*Mar 26 04:07:48.671: IP: s=1.1.1.1 (GigabitEthernet0/1), d=3.3.3.3, len 40, rcvd 4
 -à  Traffic entering from Optimizer interface

*Mar 26 04:07:48.671:     TCP src=1720, dst=80, seq=0, ack=0, win=0
```

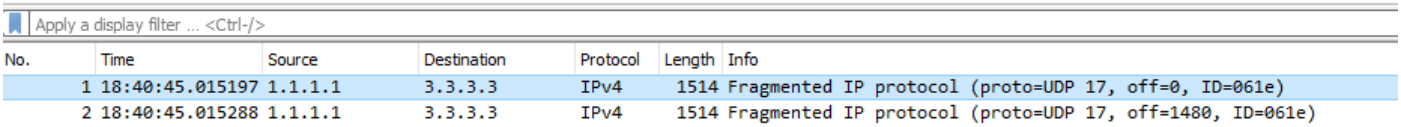
UDP packet -port 9201

```
*Mar 27 09:30:19.879: IP: s=1.1.1.1 (GigabitEthernet0/1), d=3.3.3.3, len 28, input
feature  à Traffic entering from Optimizer interface

*Mar 27 09:30:19.879:     UDP src=6000, dst=9201, MCI Check(80), rtype 0, forus FALSE,
sendself FALSE, mtu 0, fwdchk FALSE
```

Testgeval 2: Verkeer geïnitieerd via SNELbestand van LAN-router naar firewall met UDP 500

Verkeer met twee fragmenten in het volgende bestand:



1. Eerste fragmentaties met routekaart:

- Het eerste fragment met **offset = 0** is bekend als beginfragment en het bevat de UDP-header in het pakket.
- Aangezien het verkeer voor UDP 500 is, wordt het in volgorde 70 aangepast om **elke IP** toe te

staan.

```
        prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0   [23]
```

- Dus het allereerste pakket dat zowel Layer 3 als Layer 4 informatie heeft wordt correct routeerd.

2. Niet-eerste fragmentatiepakketten met routekaart:

- Het tweede fragment met **Offset ¥ 0** is bekend als niet-initieel fragment en bevat geen UDP-header. Het is een puur IP-pakket met het protocol type UDP (17).
- Aangezien er geen Layer 4-informatie is, komt deze in volgorde 70 overeen: **ip 0.0.0.0/0 0.0.0.0/0**
- Echter, in volgorde 50 is er een toegangslijst die verkeer voor UDP poort 920x aansluit. De hardware maakt automatisch een invoer aan om de UDP fragmenten toe te staan die overeenkomen met de gespecificeerde Layer 3 informatie.
- Daarom elk gefragmenteerd pakket voor om het even welke Layer 3 informatie met UDP protocol dat in volgorde 50 wordt gematcht.

```
            prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9201 flow-label 9201  [0]

            prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment [23]
>>>>>>>>>>>>>>>>>>>>
```

- Op deze manier is er één fragment dat goed wordt gerouteerd en een ander via verkeerde sequentie.
- Het tweede fragment wordt aangepast om **offset = 0** te maken en het wordt in volgorde 70 aangepast zoals verwacht.
- Dit is een verwacht gedrag wanneer Layer 4 fragmenten worden ontvangen.
- Het voornemen om een extra artikel te creëren om fragmenten toe te staan, is om de niet-initiële fragmenten toe te staan die zonder Layer 4-informatie worden ontvangen.
- In het geval was het verkeer bestemd voor UDP 9201 en was er geen toegang om fragmenten toe te staan. Dan zou het tweede fragment gelijk zijn in 70 om **elke** ip toe **te staan** en dus ten onrechte zijn routeerd.

```
Nexus# sh route-map In_to_Out pbr-statistics

route-map In_to_Out, permit, sequence 3

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 5

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 7

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 10
```

```
 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 30

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 35

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 40

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 50 ----------> 2nd Fragment for UDP 500 is matched here

 Policy routing matches: 4397 packets

route-map In_to_Out, permit, sequence 70------------> 1st Fragment for UDP 500 is matched here

 Policy routing matches: 4397 packets
```

- Een andere sequentie 45 wordt gecreëerd om het verkeer voor UDP 500 mogelijk te maken en te zien dat beide fragmenten in volgorde 45 worden geneutraliseerd.
- Het eerste fragment dat is aangepast als gevolg van UDP-headerinformatie en niet-initieel afgesloten in de fragmenten-regel voor sequentie 45.

```
Nexus# sh route-map In_to_Out pbr-statistics

route-map In_to_Out, permit, sequence 3

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 5

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 7

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 10

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 30

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 35

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 40

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 45----------> Both fragments matched here

 Policy routing matches: 213 packets
```

```
route-map In_to_Out, permit, sequence 50

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 70

 Policy routing matches: 0 packets


Default routing: 0 packets
```

Toegangslijst voor sequentie 45:

```
Nexus# sh ip access-lists udptraffic

IP access list udptraffic

permit udp any any eq isakmp
```

3. Let nu op hoe fragmenten sleutelwoord zich gedraagt met ACL en Route-Map

- Volgorde 5 wordt toegepast om elke willekeurige UDP-poort 56 op de poort ACL toe te staan.

```
   Nexus# sh ip access-lists TEST_UDP


IP access list TEST_UDP

statistics per-entry

 5 permit udp any any eq 56 [match=0]

 10 permit udp any any eq isakmp [match=0]

 20 permit ip any any [match=0]
```

- Initieerde een verkeersstroom met gefragmenteerd niet-eerste pakket en merkte het op dat het in volgorde 5 bij elkaar kwam. Zelfs al is het pakket voor UDP 500, komt het in volgorde 5 overeen om UDP 56 toe te staan.

```
Nexus# sh ip access-lists TEST_UDP


IP access list TEST_UDP

statistics per-entry

 5 permit udp any any eq 56 [match=56]

 10 permit udp any any eq isakmp [match=0]

 20 permit ip any any [match=0]
```

- De fragmenten worden ontkend op de poort-ACL en het is opgemerkt dat er geen pakketten worden gevonden in het ACL-kader voor niet-beginfase omdat het pakket in de **entry-udp**

wordt aangepast **aan fragmenten** die automatisch door platform worden gemaakt.

```
NEXUS# sh ip access-lists TEST_UDP


IP access list TEST_UDP
        statistics per-entry
         fragments deny-all
        5 permit udp any any eq 56 [match=0]
         10 permit udp any any eq isakmp [match=0]
         20 permit ip any any [match=0]


 [0014:000a:000a] prec 3 permit udp 0.0.0.0/0 0.0.0.0/0 eq 56 flow-label 56  [0]-> Here we are
now not seeing any entry to allow UDP fragments
 [0015:000b:000b] prec 3 permit udp 0.0.0.0/0 0.0.0.0/0 eq 500 flow-label 500  [0]
 [0016:000c:000c] prec 3 permit ip 0.0.0.0/0 0.0.0.0/0   [0]
 [0017:000d:000d] prec 3 deny ip 0.0.0.0/0 0.0.0.0/0 fragment   [100]>> Getting matched in
fragments deny statement
 [001e:0014:0014] prec 3 deny ip 0.0.0.0/0 0.0.0.0/0   [0]
```

- Ontkent de fragmenten in problematische ACL in PBR, maar deze werkweg werkte niet en pakketten worden nog gezien om in zowel sequentie 50 als 70 aan te passen. Dit is te wijten aan het programmeergedrag van Toegangslijst en Routemap.

```
          NEXUS# sh ip access-lists UDP_Traffic
       IP access list UDP_Traffic
       statistics per-entry
        fragments deny-all
       10 permit udp any any eq 9201
        20 permit udp any any eq 9202
        30 permit udp any any eq 9203


[0061:0057:0057] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9201 flow-label 9201
[0]
[0062:0058:0058] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment    [8027]
 [0063:0059:0059] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9202 flow-label 9202
[0]
```

```
 [0064:005a:005a] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment    [0]

 [0065:005b:005b] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9203 flow-label 9203
[0]

 [0066:005c:005c] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment    [0]

  [0067:005d:005d] prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0   [8027]

 [0068:005e:005e] prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0   [0]
```

- Uitvoer wanneer fragmenten ontkennen wordt toegepast op zowel poort-ACL als PBR-ACL:

```
[0061:0057:0057] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9201 flow-label 9201
[0]

[0062:0058:0058] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment    [8027] ---
> Once the fragments are denied in port CAL, we observed non-initial packets to be getting
dropped (See the mismatch in number of packets between UDP and IP counter)

 [0063:0059:0059] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9202 flow-label 9202
[0]

 [0064:005a:005a] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment    [0]

 [0065:005b:005b] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 eq 9203 flow-label 9203
[0]

 [0066:005c:005c] prec 1 redirect(0x5e)-routed udp 0.0.0.0/0 0.0.0.0/0 fragment    [0]

  [0067:005d:005d] prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0   [8214]

 [0068:005e:005e] prec 1 permit-routed ip 0.0.0.0/0 0.0.0.0/0   [0]



VDC-1 Ethernet2/1 :

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

INSTANCE 0x0

---------------

 Tcam 0 resource usage:

 ----------------------

 Label_a = 0x200

  Bank 0

  ------

    IPv4 Class

      Policies: PACL(TEST_UDP)

       Netflow profile: 0

      Netflow deny profile: 0
```

```
      Entries:

         [Index] Entry [Stats]

        ---------------------

 [0014:000a:000a] prec 3 permit udp 0.0.0.0/0 0.0.0.0/0 eq 56 flow-label 56  [8027]

 [0015:000b:000b] prec 3 permit udp 0.0.0.0/0 0.0.0.0/0 eq 500 flow-label 500  [8214]

 [0016:000c:000c] prec 3 permit ip 0.0.0.0/0 0.0.0.0/0   [0]

 [0017:000d:000d] prec 3 deny ip 0.0.0.0/0 0.0.0.0/0 fragment   [100]

 [001e:0014:0014] prec 3 deny ip 0.0.0.0/0 0.0.0.0/0   [0]
```

Er zijn verschillende manieren om dit probleem of de beperking van gefragmenteerde pakketten met L4-informatie te overwinnen:

- Routekaart kan worden aangepast om specifieke L3-informatie voor bepaalde UDP-poorten mogelijk te maken.

In de huidige configuratie, als L3 bron- en doelinformatie wordt vermeld dan wordt het niet-initiële pakket op basis van die specifieke informatie routeerd. Dit is echter alleen bruikbaar als er geen andere sequentie is voordat deze overeenkomt met dezelfde L3-informatie.

```
Nexus# show ip access-lists UDP_Traffic

IP access list UDP_Traffic

10 permit udp host 1.1.1.1 host 3.3.3.3 eq 9201
20 permit udp any any eq 9202

30 permit udp any any eq 9203
```

- Het pad van bron naar bestemming kan worden geverifieerd om de MTU te controleren zodat het pakket niet gefragmenteerd wordt.
- Het toepassen van een andere reeks maakt het mogelijk dat UDP boven de problematische sequentie werkt, maar het gedrag is hetzelfde als eerder werd uitgelegd toen sequentie 45 werd toegepast

```
Nexus# sh route-map In_to_Out pbr-statistics

route-map In_to_Out, permit, sequence 3

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 5

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 7

 Policy routing matches: 0 packets
```

```
route-map In_to_Out, permit, sequence 10

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 30

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 35

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 40

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 45----------> Both fragments matched here

 Policy routing matches: 213 packets

route-map In_to_Out, permit, sequence 50

 Policy routing matches: 0 packets

route-map In_to_Out, permit, sequence 70

 Policy routing matches: 0 packets
```

Toegangslijst voor sequentie 45:

```
Nexus# sh ip access-lists udptraffic
```

IP-toegangslijst voor verkeer:

```
permit udp any any eq isakmp
```

Doc Bug: CSCve05428 N7K Doc bug | ACL in PBR die zowel L3- als L4-informatie bevat.