

# Catalyst 9000 스위치에서 NAT 구성 및 확인

## 목차

---

- [소개](#)
- [사전 요구 사항](#)
  - [요구 사항](#)
- [배경 정보](#)
  - [사용되는 구성 요소](#)
- [용어](#)
- [네트워크 다이어그램](#)
- [구성](#)
  - [컨피그레이션 예](#)
- [고정 NAT 확인](#)
  - [소프트웨어 확인](#)
  - [하드웨어 확인](#)
- [동적 NAT 확인](#)
  - [소프트웨어 확인](#)
  - [하드웨어 확인](#)
- [PAT\(Dynamic NAT Overload\) 확인](#)
  - [소프트웨어 확인](#)
  - [하드웨어 확인](#)
- [패킷 레벨 디버그](#)
- [NAT 확장 문제 해결](#)
  - [AOT\(Address Only Translation\)](#)
- [관련 정보](#)

---

## 소개

이 문서에서는 Catalyst 9000 플랫폼에서 NAT(Network Address Translation)를 구성하고 검증하는 방법에 대해 설명합니다.

## 사전 요구 사항

### 요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- IP 주소 지정
- 액세스 제어 목록

## 배경 정보

NAT의 가장 일반적인 경우는 사설 IP 네트워크 공간을 전역 고유 인터넷 라우팅 가능 주소로 변환하는 데 사용됩니다.

NAT를 수행하는 디바이스에는 내부 네트워크(로컬)의 인터페이스와 외부 네트워크(글로벌)의 인터페이스가 필요합니다.

NAT 장치는 NAT 규칙 컨피그레이션을 기반으로 변환이 필요한지 여부를 확인하기 위해 소스 트래픽의 검사를 담당합니다.

변환이 필요한 경우 디바이스는 로컬 소스 IP 주소를 전역 고유 IP 주소로 변환하고 NAT 변환 테이블에서 이를 추적합니다.

패킷이 라우팅 가능한 주소로 다시 들어오면 디바이스는 NAT 테이블을 검사하여 다른 변환이 제대로 이루어졌는지 확인합니다.

이 경우 라우터는 내부 전역 주소를 다시 적절한 내부 로컬 주소로 변환하고 패킷을 라우팅합니다.

### 사용되는 구성 요소

Cisco IOS® XE 16.12.1 NAT를 통해 이제 Network Advantage 라이선스를 사용할 수 있습니다. 이전의 모든 릴리스에서는 DNA Advantage 라이선스에서 사용할 수 있습니다.

플랫폼	NAT 기능 도입
C9300	Cisco IOS® XE 버전 16.10.1
C9400	Cisco IOS® XE 버전 17.1.1
C9500	Cisco IOS® XE 버전 16.5.1a
C9600	Cisco IOS® XE 버전 16.11.1

이 문서는 Cisco IOS® XE 버전 16.12.4를 사용하는 Catalyst 9300 플랫폼을 기반으로 합니다

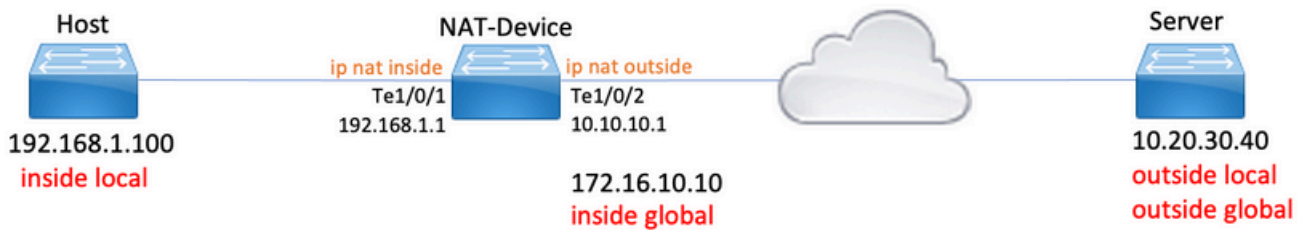
이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

### 용어

고정 NAT	로컬 주소를 전역 주소에 1:1 매핑할 수 있습니다.
동적 NAT	로컬 주소를 전역 주소 풀에 매핑합니다.
오버로드 NAT	로컬 주소를 고유한 L4 포트를 사용하는 단일 전역 주소에 매핑합니다.
내부 로컬	내부 네트워크의 호스트에 할당된 IP 주소.
내부 글로벌	외부 네트워크에 나타나는 내부 호스트의 IP 주소입니다. 이를 내부 로컬이 번역된 주소로 생각하면 됩니다.

외부 로컬	내부 네트워크에 나타나는 외부 호스트의 IP 주소.
외부 전역	외부 네트워크의 호스트에 할당되는 IP 주소. 대부분의 경우 외부 로컬 주소와 외부 글로벌 주소는 동일합니다.
FMAN-RP	기능 관리자 RP. 프로그래밍 정보를 FMAN-FP에 전달하는 Cisco IOS® XE의 제어 플레인입니다.
FMAN-FP	기능 관리자 FP FMAN-FP는 FMAN-RP로부터 정보를 받아서 FED로 전달한다.
연방	포워딩 엔진 드라이버. FMAN-FP는 FED를 사용하여 제어 평면에서 UADP(Unified Access Data Plane) ASIC(Application Specific Integrated Circuit)로 정보를 프로그래밍합니다.

## 네트워크 다이어그램



## 구성

### 컨피그레이션 예

192.168.1.100(내부 로컬)을 172.16.10.10(내부 전역)으로 변환하는 고정 NAT 컨피그레이션:

```
<#root>
```

```
NAT-Device#
```

```
show run interface te1/0/1
```

```
Building configuration...
```

```
Current configuration : 109 bytes
```

```
!
```

```
interface TenGigabitEthernet1/0/1
```

```
no switchport
```

```
ip address 192.168.1.1 255.255.255.0
```

```
ip nat inside
```

```
<-- NAT inside interface
```

```
end
```



NAT-Device#

show run interface te1/0/2

Building configuration...

Current configuration : 109 bytes

```
!  
interface TenGigabitEthernet1/0/2  
no switchport  
ip address 10.10.10.1 255.255.255.0  
  
ip nat outside
```

<-- NAT outside interface

```
end  
!
```

```
ip nat pool TAC-POOL 172.16.10.1 172.16.10.30 netmask 255.255.255.224 <-- NAT pool configuration
```

```
ip nat inside source list hosts pool TAC-POOL
```

<-- NAT rule configuration

```
!
```

```
ip access-list standard hosts <-- ACL to match hosts to be
```

```
10 permit 192.168.1.0 0.0.0.255
```

NAT-Device#

show ip nat translations

Pro	Inside global	Inside local	Outside local	Outside global
icmp	172.16.10.10:6	192.168.1.100:6	10.20.30.40:6	10.20.30.40:6
---	172.16.10.10	192.168.1.100	---	---

192.168.1.0/24을 10.10.10.1(ip nat 외부 인터페이스)로 변환하는 PAT(Dynamic NAT Overload) 컨피그레이션:

<#root>

NAT-Device#

show run interface te1/0/1

Building configuration...

Current configuration : 109 bytes

```

!
interface TenGigabitEthernet1/0/1
no switchport
ip address 192.168.1.1 255.255.255.0

ip nat inside                                <-- NAT inside interface

end

NAT-Device#
show run interface te1/0/2

Building configuration...

Current configuration : 109 bytes
!
interface TenGigabitEthernet1/0/2
no switchport
ip address 10.10.10.1 255.255.255.0

ip nat outside                                <-- NAT outside interface

end
!
ip nat inside source list hosts interface TenGigabitEthernet1/0/2 overload          <-- NAT configurati

!
ip access-list standard hosts                                                        <-- ACL to match hos

10 permit 192.168.1.0 0.0.0.255

```

각 변환에 대해 내부 전역 주소의 포트가 1씩 증가합니다.

```
<#root>
```

```
NAT-Device#
```

```
show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
icmp	10.10.10.1:1024	192.168.1.100:1	10.20.30.40:1	10.20.30.40:1024

```
<-- Notice layer 4 port increments
```

icmp	10.10.10.1:1025	192.168.1.100:2	10.20.30.40:2	10.20.30.40:1025
------	-----------------	-----------------	---------------	------------------

```
<-- Notice layer 4 port increments
```

```
icmp 10.10.10.1:1026 192.168.1.100:3 10.20.30.40:3 10.20.30.40:1026
icmp 10.10.10.1:1027 192.168.1.100:4 10.20.30.40:4 10.20.30.40:1027
icmp 10.10.10.1:1028 192.168.1.100:5 10.20.30.40:5 10.20.30.40:1028
icmp 10.10.10.1:1029 192.168.1.100:6 10.20.30.40:6 10.20.30.40:1029
icmp 10.10.10.1:1030 192.168.1.100:7 10.20.30.40:7 10.20.30.40:1030
icmp 10.10.10.1:1031 192.168.1.100:8 10.20.30.40:8 10.20.30.40:1031
```

```
10.10.10.1:1024 = inside global
```

```
192.168.1.100:1 = inside local
```

## 고정 NAT 확인

### 소프트웨어 확인

변환된 활성 흐름이 없을 경우 고정 NAT가 있는 변환의 절반이 표시될 것으로 예상됩니다. 흐름이 활성화되면 동적 변환이 생성됩니다

```
<#root>
```

```
NAT-Device#
```

```
show ip nat translations
```

```
Pro Inside global      Inside local      Outside local      Outside global
icmp 172.16.10.10:10  192.168.1.100:10 10.20.30.40:10    10.20.30.40:10
```

```
<-- dynamic translation
```

```
--- 172.16.10.10      192.168.1.100      ---      ---
```

```
<-- static configuration from NAT rule configuration
```

show ip nat translations verbose 명령을 사용하면 플로우가 생성된 시간과 변환에 남아 있는 시간을 확인할 수 있습니다.

```
<#root>
```

```
NAT-Device#
```

```
show ip nat translations verbose
```

```
Pro Inside global Inside local Outside local Outside global
icmp 172.16.10.10:10 192.168.1.100:10 10.20.30.40:10 10.20.30.40:10

create 00:00:13, use 00:00:13, left 00:00:46,
```

```
<-- NAT timers
```

```
flags:
extended, use_count: 0, entry-id: 10, lc_entries: 0
--- 172.16.10.10 192.168.1.100 --- ---
create 00:09:47, use 00:00:13,
flags:
static, use_count: 1, entry-id: 9, lc_entries: 0
```

NAT 통계를 확인합니다. NAT 히트 카운터는 플로우가 NAT 규칙과 일치하고 생성될 때 증가합니다.

트래픽이 규칙과 일치하지만 변환을 생성할 수 없는 경우 NAT 실패 카운터가 증가합니다.

```
<#root>
```

```
NAT-DEVICE#
```

```
show ip nat statistics
```

```
Total active translations: 1 (
```

```
1 static,
```

```
0 dynamic; 0 extended)
```

```
<-- 1 static translation
```

```
Outside interfaces:
```

```
TenGigabitEthernet1/0/1 <-- NAT outside interface
```

```
Inside interfaces:
```

```
TenGigabitEthernet1/0/2 <-- NAT inside interface
```

```
Hits: 0 Misses: 0 <-- NAT hit and miss counters.
```

```
CEF Translated packets: 0, CEF Punted packets: 0
```

```
Expired translations: 0
```

```
Dynamic mappings:
```

```
-- Inside Source
```

```
[Id: 1] access-list hosts interface TenGigabitEthernet1/0/1 refcount 0
```

변환이 발생하려면 NAT 흐름의 소스 및 목적지에 인접해야 합니다. 인접성 ID를 기록해 둡니다.



<#root>

NAT-Device#

show ip route 10.20.30.40

Routing entry for 10.20.30.40/32  
Known via "static", distance 1, metric 0  
Routing Descriptor Blocks:  
\* 10.10.10.2  
Route metric is 0, traffic share count is 1

NAT-Device#

show platform software adjacency switch active f0

Adjacency id:

0x29(41)

<-- adjacency ID

Interface: TenGigabitEthernet1/0/1, IF index: 52, Link Type: MCP\_LINK\_IP  
Encap: 0:ca:e5:27:3f:e4:70:1f:53:0:b8:e4:8:0  
Encap Length: 14, Encap Type: MCP\_ET\_ARPA, MTU: 1500  
Flags: no-l3-inject  
Incomplete behavior type: None  
Fixup: unknown  
Fixup\_Flags\_2: unknown  
Nexthop addr:  
192.168.1.100

<-- source adjacency

IP FRR MCP\_ADJ\_IPFRR\_NONE 0  
aom id: 464, HW handle: (nil) (created)

Adjacency id:

0x24 (36)

<-- adjacency ID


Interface: TenGigabitEthernet1/0/2, IF index: 53, Link Type: MCP\_LINK\_IP  
Encap: 34:db:fd:ee:ce:e4:70:1f:53:0:b8:d6:8:0  
Encap Length: 14, Encap Type: MCP\_ET\_ARPA, MTU: 1500  
Flags: no-l3-inject  
Incomplete behavior type: None  
Fixup: unknown  
Fixup\_Flags\_2: unknown  
Nexthop addr:  
10.10.10.2

```
<-- next hop to 10.20.30.40
```

```
IP FRR MCP_ADJ_IPFRR_NONE 0  
aom id: 452, HW handle: (nil) (created)
```

스위치에서 트래픽을 수신하고 스위치에서 NAT 플로우를 생성하는지 확인하기 위해 NAT 디버그를 활성화할 수 있습니다

---

 참고: NAT의 대상이 되는 ICMP 트래픽은 항상 소프트웨어에서 처리되므로 플랫폼 디버그가 ICMP 트래픽에 대한 로그를 표시하지 않습니다.

---

```
<#root>
```

```
NAT-Device#
```

```
debug ip nat detailed
```

```
IP NAT detailed debugging is on
```

```
NAT-Device#
```

```
*Mar 8 23:48:25.672: NAT: Entry assigned id 11
```

```
<-- receive traffic and flow created
```

```
*Mar 8 23:48:25.672: NAT: i: icmp (192.168.1.100, 11) -> (10.20.30.40, 11) [55]
```

```
*Mar 8 23:48:25.672: NAT:
```

```
s=192.168.1.100->172.16.10.10
```

```
, d=10.20.30.40 [55]NAT: dyn flow info download suppressed for flow 11
```

```
<-- source is translated
```

```
*Mar 8 23:48:25.673: NAT: o: icmp (10.20.30.40, 11) -> (172.16.10.10, 11) [55]
```

```
*Mar 8 23:48:25.674: NAT: s=10.20.30.40,
```

```
d=172.16.10.10->192.168.1.100
```

```
[55]NAT: dyn flow info download suppressed for flow 11
```

```
<-- return source is translated
```

```
*Mar 8 23:48:25.675: NAT: i: icmp (192.168.1.100, 11) -> (10.20.30.40, 11) [56]
```

플로우가 만료되거나 삭제되면 디버그에 DELETE 작업이 표시됩니다.

<#root>

\*Mar 31 17:58:31.344: FMANRP-NAT: Received flow data, action:

DELETE

<-- action is delete

\*Mar 31 17:58:31.344: id 2, flags 0x1, domain 0
src\_local\_addr 192.168.1.100, src\_global\_addr 172.16.10.10, dst\_local\_addr 10.20.30.40,
dst\_global\_addr 10.20.30.40, src\_local\_port 31783, src\_global\_port 31783,
dst\_local\_port 23, dst\_global\_port 23,
proto 6, table\_id 0 inside\_mapping\_id 0,
outside\_mapping\_id 0, inside\_mapping\_type 0,
outside\_mapping\_type 0

### 하드웨어 확인

NAT 규칙이 구성된 경우 디바이스는 NAT Region 5(NAT 영역 5) 아래의 TCAM에서 이 규칙을 프로그래밍합니다. 규칙이 TCAM에 프로그래밍되었는지 확인합니다.

출력은 16진수이므로 IP 주소로 변환해야 합니다.

<#root>

NAT-Device#

show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT

Printing entries for region NAT\_1 (370) type 6 asic 3

Printing entries for region NAT\_2 (371) type 6 asic 3

Printing entries for region NAT\_3 (372) type 6 asic 3

Printing entries for region NAT\_4 (373) type 6 asic 3

Printing entries for region NAT\_5 (374) type 6 asic 3

<-- NAT Region 5

TAQ-2 Index-128 (A:1,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0
Mask1 3300f000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:ffffffffff
Key1 21009000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:

c0a80164

<--

inside local IP address 192.168.1.100 in hex (c0a80164)

AD 10087000:00000073

TAQ-2 Index-129 (A:1,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0  
Mask1 0300f000:00000000:00000000:00000000:00000000:00000000:ffffff:00000000  
Key1 02009000:00000000:00000000:00000000:00000000:00000000:

ac100a0a

:00000000

<-- inside global IP address 172.16.10.10 in hex (ac100a0a)

AD 10087000:00000073

마지막으로, 플로우가 활성화되면 NAT Region 1 아래에서 TCAM의 확인을 통해 하드웨어 프로그래밍을 확인할 수 있습니다.

<#root>

NAT-Device#

show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT

Printing entries for region

NAT\_1

(370) type 6 asic 1

<-- NAT Region 1

=====  
TAQ-2 Index-32 (A:0,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0  
Mask1 0000f000:ff00ffff:00000000:0000ffff:00000000:00000000:ffffff:ffffff  
Key1 00009000:06005ac9:00000000:00000017:00000000:00000000:

0a141e28:c0a80164

AD 10087000:000000b0

TAQ-2 Index-33 (A:0,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0  
Mask1 0000f000:ff00ffff:00000000:0000ffff:00000000:00000000:ffffff:ffffff  
Key1 00009000:06000017:00000000:00005ac9:00000000:00000000:

ac100a0a:0a141e28

AD 10087000:000000b1

Starting at Index-32 Key1 from right to left:

c0a80164

= 192.168.1.100 (Inside Local)

0a141e28

= 10.20.30.40 (Outside Global)

00000017

= 23 (TCP destination port)

06005ac9

= 06 for TCP and 5ac9 is 23241 which is source port from "show ip nat translations" of the inside host

Repeat the same for Index-33 which is the reverse translation:

0a141e28

= 10.20.30.40 (Outside Global)

ac100a0a

= 172.16.10.10 (Inside Global)

00005ac9

= 23241 TCP Destination port

06000017

= 06 for TCP and 17 for TCP source port 23

## 동적 NAT 확인

### 소프트웨어 확인

내부 IP 주소를 (으)로 변환할 주소 풀이 구성되었는지 확인합니다.

이 컨피그레이션을 통해 192.168.1.0/24 네트워크를 주소 172.16.10.1에서 172.16.10.254로 변환할 수 있습니다

```
<#root>
```

```
NAT-Device#
```

```
show run | i ip nat
```

```
ip nat inside
```

```
<-- ip nat inside on inside interface
```

```
ip nat outside
```

```
<-- ip nat outside on outside interface
```

```
ip nat pool MYPOOL 172.16.10.1 172.16.10.254 netmask 255.255.255.0 <-- Pool of addresses to translate

ip nat inside source list hosts pool MYPOOL <-- Enables hosts that match ACL "1
```

NAT-Device#

```
show ip access-list 10 <-- ACL to match hosts to be translated
```

```
Standard IP access list 10
10 permit 192.168.1.0, wildcard bits 0.0.0.255
NAT-Device#
```

동적 NAT에서는 컨피그레이션만 있는 엔트리를 생성하지 않습니다. 변환 테이블을 채우기 전에 활성 흐름을 만들어야 합니다.

<#root>

NAT-Device#

```
show ip nat translations
```

<...empty...>

NAT 통계를 확인합니다. NAT 히트 카운터는 플로우가 NAT 규칙과 일치하고 생성될 때 증가합니다.

트래픽이 규칙과 일치하지만 변환을 생성할 수 없는 경우 NAT 실패 카운터가 증가합니다.

<#root>

NAT-DEVICE#

```
show ip nat statistics
```

```
Total active translations: 3794 (1 static,
```

```
3793 dynamic
```

```
; 3793 extended)
```

```
<-- dynamic translations
```

Outside interfaces:

```
TenGigabitEthernet1/0/1 <-- NAT outside interface
```

Inside interfaces:

```
TenGigabitEthernet1/0/2          <-- NAT inside interface

Hits: 3793
  Misses: 0
<-- 3793 hits

CEF Translated packets: 0, CEF Punted packets: 0
Expired translations: 0

Dynamic mappings:                  <-- rule for dynamic mappings

-- Inside Source
[Id: 1]

access-list hosts interface TenGigabitEthernet1/0/1

  refcount 3793
<-- NAT rule displayed
```

소스 및 대상에 대한 인접성이 있는지 확인합니다.

```
<#root>
```

```
NAT-Device#
```

```
show platform software adjacency switch active f0
```

```
Number of adjacency objects: 4
```

```
Adjacency id:
```

```
0x24(36)
```

```
<-- adjacency ID
```

```
Interface: TenGigabitEthernet1/0/2, IF index: 53, Link Type: MCP_LINK_IP
```

```
Encap: 34:db:fd:ee:ce:e4:70:1f:53:0:b8:d6:8:0
```

```
Encap Length: 14, Encap Type: MCP_ET_ARPA, MTU: 1500
```

```
Flags: no-l3-inject
```

```
Incomplete behavior type: None
```

```
Fixup: unknown
```

```
Fixup_Flags_2: unknown
```

```
Nexthop addr:
```

```
10.10.10.2
```

```
<-- adjacency to destination
```

```
IP FRR MCP_ADJ_IPFRR_NONE 0
```

aom id: 449, HW handle: (nil) (created)

Adjacency id:

0x25 (37)

<-- adjacency ID

Interface: TenGigabitEthernet1/0/1, IF index: 52, Link Type: MCP\_LINK\_IP

Encap: 0:ca:e5:27:3f:e4:70:1f:53:0:b8:e4:8:0

Encap Length: 14, Encap Type: MCP\_ET\_ARPA, MTU: 1500

Flags: no-l3-inject

Incomplete behavior type: None

Fixup: unknown

Fixup\_Flags\_2: unknown

Nexthop addr:

192.168.1.100

<-- source adjacency

IP FRR MCP\_ADJ\_IPFRR\_NONE 0

aom id: 451, HW handle: (nil) (created)

NAT 문제가 있는 경우 인접성이 확인된 후 플랫폼 독립적인 NAT 디버그로 시작할 수 있습니다

<#root>

NAT-Device#

debug ip nat

IP NAT debugging is on

NAT-Device#

debug ip nat detailed

IP NAT detailed debugging is on

NAT-Device#

show logging

\*May 13 01:00:41.136: NAT: Entry assigned id 6

\*May 13 01:00:41.136: NAT: Entry assigned id 7

\*May 13 01:00:41.136: NAT: i:

tcp (192.168.1.100, 48308)

-> (10.20.30.40, 23) [30067]

<-- first packet ingress without NAT



```

*May 13 01:00:41.136: NAT: TCP Check for Limited ALG Support
*May 13 01:00:41.136: NAT:
s=192.168.1.100->172.16.10.10
, d=10.20.30.40 [30067]NAT: dyn flow info download suppressed for flow 7
<-- confirms source address translation

*May 13 01:00:41.136: NAT: attempting to setup alias for 172.16.10.10 (redundancy_name , idb NULL, flag
*May 13 01:00:41.139: NAT: o:
tcp (10.20.30.40, 23)
-> (172.16.10.10, 48308) [40691]
<-- return packet from destination to be translated

*May 13 01:00:41.139: NAT: TCP Check for Limited ALG Support
*May 13 01:00:41.139: NAT: s=10.20.30.40,
d=172.16.10.10->192.168.1.100
[40691]NAT: dyn flow info download suppressed for flow 7
<-- return packet is translated

*May 13 01:00:41.140: NAT: i: tcp (192.168.1.100, 48308) -> (10.20.30.40, 23) [30068]

```

또한 FMAN-RP NAT 작업을 디버깅할 수 있습니다.

```
<#root>
```

```
NAT-Device#
```

```
debug platform software nat all
```

```
NAT platform all events debugging is on
```

```
Log Buffer (100000 bytes):
```

```
*May 13 01:04:16.098: FMANRP-NAT: Received flow data, action:
```

```
ADD
```

```
<-- first packet in flow so we ADD an entry
```

```
*May 13 01:04:16.098: id 9, flags 0x1, domain 0
```

```
src_local_addr 192.168.1.100, src_global_addr 172.16.10.10, dst_local_addr 10.20.30.40
```

```
,
```

```
<-- verify inside local/global and outside local/global
```

```
dst_global_addr 10.20.30.40, src_local_port 32529, src_global_port 32529,
dst_local_port 23, dst_global_port 23
,
<-- confirm ports, in this case they are for Telnet
```

```
proto 6, table_id 0 inside_mapping_id 1,
outside_mapping_id 0, inside_mapping_type 2,
outside_mapping_type 0
*May 13 01:04:16.098: FMANRP-NAT: Created TDL message for flow info:
ADD id 9
*May 13 01:04:16.098: FMANRP-NAT: Sent TDL message for flow data config:
ADD id 9
```

```
*May 13 01:04:16.098: FMANRP-NAT: Received flow data, action:
MODIFY          <-- subsequent packets are MODIFY
```

```
*May 13 01:04:16.098: id 9, flags 0x1, domain 0
src_local_addr 192.168.1.100, src_global_addr 172.16.10.10, dst_local_addr 10.20.30.40,
dst_global_addr 10.20.30.40, src_local_port 32529, src_global_port 32529,
dst_local_port 23, dst_global_port 23,
proto 6, table_id 0 inside_mapping_id 1,
outside_mapping_id 0, inside_mapping_type 2,
outside_mapping_type 0
*May 13 01:04:16.098: FMANRP-NAT: Created TDL message for flow info:
MODIFY id 9
*May 13 01:04:16.098: FMANRP-NAT: Sent TDL message for flow data config:
MODIFY id 9
```

만료 또는 수동 제거와 같은 이유로 규칙이 제거되는 경우 DELETE 작업이 관찰됩니다.

<#root>

```
*May 13 01:05:20.276: FMANRP-NAT: Received flow data, action:
DELETE          <-- DELETE action
```

```
*May 13 01:05:20.276: id 9, flags 0x1, domain 0
src_local_addr 192.168.1.100, src_global_addr 172.16.10.10, dst_local_addr 10.20.30.40,
dst_global_addr 10.20.30.40, src_local_port 32529, src_global_port 32529,
dst_local_port 23, dst_global_port 23,
proto 6, table_id 0 inside_mapping_id 0,
outside_mapping_id 0, inside_mapping_type 0,
outside_mapping_type 0
```

## 하드웨어 확인

변환할 트래픽과 일치하는 NAT 규칙이 NAT 영역 5 아래의 하드웨어에 올바르게 추가되었는지 확인합니다.

```
<#root>
```

```
NAT-Device#
```

```
show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT_
```

```
Printing entries for region
```

```
NAT_1
```

```
(370) type 6 asic 1
```

```
<<<< empty due to no active flow
```

```
=====  
Printing entries for region NAT_2 (371) type 6 asic 1  
=====
```

```
Printing entries for region NAT_3 (372) type 6 asic 1  
=====
```

```
Printing entries for region NAT_4 (373) type 6 asic 1  
=====
```

```
Printing entries for region NAT_5 (374) type 6 asic 1  
=====
```

```
TAQ-2 Index-128 (A:0,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0  
Mask1 0300f000:00000000:00000000:00000000:00000000:00000000:ffffff8:00000000  
Key1 02009000:00000000:00000000:00000000:00000000:00000000:ac100a00:00000000  
AD 10087000:00000073
```

```
TAQ-2 Index-129 (A:0,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0  
Mask1 3300f000:00000000:00000000:00000000:00000000:00000000:00000000:
```

```
ffffff00
```

```
Key1 21009000:00000000:00000000:00000000:00000000:00000000:00000000:
```

```
c0a80100
```

```
AD 10087000:00000073
```

```
ffffff00 = 255.255.255.0 in hex
```

```
c0a80100 = 192.168.1.0 in hex which matches our network in the NAT ACL
```

마지막으로, NAT TCAM Region 1에서 활성 변환이 올바르게 프로그래밍되었는지 확인해야 합니다

```
<#root>
```

```
NAT-Device#
```

```
show ip nat translations
```

```
Pro Inside global      Inside local      Outside local     Outside global  
tcp 172.16.10.10:54854 192.168.1.100:54854 10.20.30.40:23   10.20.30.40:23
```

--- 172.16.10.10            192.168.1.100            ---            ---

NAT-Device#

show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT\_

Printing entries for region

NAT\_1

(370) type 6 asic 1

=====  
TAQ-2 Index-32 (A:0,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0  
Mask1 0000f000:ff00ffff:00000000:0000ffff:00000000:00000000:ffffffff:ffffffff  
Key1 00009000:0600d646:00000000:00000017:00000000:00000000:

0a141e28

:

c0a80164

AD 10087000:000000b0

TAQ-2 Index-33 (A:0,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0  
Mask1 0000f000:ff00ffff:00000000:0000ffff:00000000:00000000:ffffffff:ffffffff  
Key1 00009000:06000017:00000000:0000d646:00000000:00000000:

ac100a0a

:

0a141e28

AD 10087000:000000b1

Printing entries for region NAT\_2 (371) type 6 asic 1

=====  
Printing entries for region NAT\_3 (372) type 6 asic 1

=====  
Printing entries for region NAT\_4 (373) type 6 asic 1

=====  
Printing entries for region NAT\_5 (374) type 6 asic 1

Starting at Index-32 Key 1 from right to left:

c0a80164

- 192.168.1.100 (inside local)

0a141e28

- 10.20.30.40 (outside local/global)

00000017

- TCP port 23

0600d646

- 6 for TCP protocol and 54854 for TCP source port

Starting at Index-33 Key 1 from right to left

0a141e28

- 10.20.30.40 destination address

ac100a0a

- 172.16.10.10 (inside global source IP address)

0000d646

- TCP source port

06000017

- TCP protocol 6 and 23 for the TCP destination port

## PAT(Dynamic NAT Overload) 확인

### 소프트웨어 확인

PAT를 확인하는 로그 프로세스는 동적 NAT와 동일합니다. 올바른 포트 변환 및 하드웨어에 포트가 올바르게 프로그래밍되었는지 확인하기만 하면 됩니다.

PAT는 NAT 규칙에 추가된 "overload" 키워드에 의해 수행됩니다.

```
<#root>
```

```
NAT-Device#
```

```
show run | i ip nat
```

```
ip nat inside
```

```
<-- ip nat inside on NAT inside interface
```

```
ip nat outside
```

```
<-- ip nat outside on NAT outside interface
```

```
ip nat pool MYPOOL 172.16.10.1 172.16.10.254 netmask 255.255.255.0 <-- Address pool to translate to
```

```
ip nat inside source list hosts pool MYPOOL overload <-- Links ACL hosts to address pool
```

소스 및 대상에 대한 인접성이 있는지 확인합니다.

<#root>

NAT-Device#

show ip route 10.20.30.40

Routing entry for 10.20.30.40/32  
Known via "static", distance 1, metric 0  
Routing Descriptor Blocks:  
\*

10.10.10.2

Route metric is 0, traffic share count is 1

NAT-Device#

show platform software adjacency switch active f0

Number of adjacency objects: 4

Adjacency id:

0x24

(36)

<-- adjacency ID

Interface: TenGigabitEthernet1/0/2, IF index: 53, Link Type: MCP\_LINK\_IP  
Encap: 34:db:fd:ee:ce:e4:70:1f:53:0:b8:d6:8:0  
Encap Length: 14, Encap Type: MCP\_ET\_ARPA, MTU: 1500  
Flags: no-l3-inject  
Incomplete behavior type: None  
Fixup: unknown  
Fixup\_Flags\_2: unknown  
Nexthop addr:

10.10.10.2 <-- adjacency to destination

IP FRR MCP\_ADJ\_IPFRR\_NONE 0  
aom id: 449, HW handle: (nil) (created)

Adjacency id:

0x25

(37)

<-- adjacency ID

```
Interface: TenGigabitEthernet1/0/1, IF index: 52, Link Type: MCP_LINK_IP
Encap: 0:ca:e5:27:3f:e4:70:1f:53:0:b8:e4:8:0
Encap Length: 14, Encap Type: MCP_ET_ARPA, MTU: 1500
Flags: no-l3-inject
Incomplete behavior type: None
Fixup: unknown
Fixup_Flags_2: unknown
Nexthop addr:
192.168.1.100          <-- source adjacency
```

```
IP FRR MCP_ADJ_IPFRR_NONE 0
aom id: 451, HW handle: (nil) (created)
```

흐름이 활성 상태일 때 변환 테이블에 변환이 추가되었는지 확인합니다. PAT를 사용하면 동적 NAT와 마찬가지로 절반 항목이 생성되지 않습니다.

내부 로컬 및 내부 전역 주소의 포트 번호를 추적합니다.

```
<#root>
```

```
NAT-Device#
```

```
show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
tcp	172.16.10.10:1024	192.168.1.100:52448	10.20.30.40:23	10.20.30.40:23

NAT 통계를 확인합니다. NAT 히트 카운터는 플로우가 NAT 규칙과 일치하고 생성될 때 증가합니다.

트래픽이 규칙과 일치하지만 변환을 생성할 수 없는 경우 NAT 실패 카운터가 증가합니다.

```
<#root>
```

```
NAT-DEVICE#
```

```
show ip nat statistics
```

```
Total active translations: 3794 (1 static,
3793 dynamic
; 3793 extended)
```

```
<-- dynamic translations
```

```
Outside interfaces:
```

```
TenGigabitEthernet1/0/1
```

```
<-- NAT outside interface
```

Inside interfaces:

TenGigabitEthernet1/0/2

<-- NAT inside interface

Hits: 3793

Misses: 0

<-- 3793 hits

CEF Translated packets: 0, CEF Punted packets: 0

Expired translations: 0

Dynamic mappings:

<-- rule for dynamic mappings

-- Inside Source

[Id: 1]

access-list hosts interface TenGigabitEthernet1/0/1

refcount 3793

<-- NAT rule displayed

Platform Independent NAT 디버깅은 포트 변환이 발생함을 보여줍니다.

<#root>

NAT-Device#

debug ip nat detailed

IP NAT detailed debugging is on

NAT-Device#

debug ip nat

IP NAT debugging is on

NAT-device#

show logging

Log Buffer (100000 bytes):

\*May 18 23:52:20.296: NAT: address not stolen for 192.168.1.100, proto 6 port 52448

\*May 18 23:52:20.296: NAT: Created portlist for proto tcp globaladdr 172.16.10.10

\*May 18 23:52:20.296: NAT: Allocated Port for 192.168.1.100 -> 172.16.10.10:

wanted 52448 got 1024<-- confirms PAT is used



```
*May 18 23:52:20.296: NAT: Entry assigned id 5
*May 18 23:52:20.296: NAT: i: tcp (192.168.1.100, 52448) -> (10.20.30.40, 23) [63338]
*May 18 23:52:20.296: NAT: TCP Check for Limited ALG Support
*May 18 23:52:20.296: NAT: TCP
```

```
s=52448->1024
```

```
, d=23
```

```
<-- confirms NAT overload with PAT
```

```
*May 18 23:52:20.296: NAT:
```

```
s=192.168.1.100->172.16.10.10, d=10.20.30.40
```

```
[63338]NAT: dyn flow info download suppressed for flow 5
```

```
<-- shows inside translation
```

```
*May 18 23:52:20.297: NAT: attempting to setup alias for 172.16.10.10 (redundancy_name , idb NULL, flag
```

```
*May 18 23:52:20.299: NAT: o: tcp (10.20.30.40, 23) -> (172.16.10.10, 1024) [55748]
```

```
*May 18 23:52:20.299: NAT: TCP Check for Limited ALG Support
```

```
*May 18 23:52:20.299: NAT: TCP s=23,
```

```
d=1024->52448
```

```
<-- shows PAT on return traffic
```

```
*May 18 23:52:20.299: NAT: s=10.20.30.40, d=172.16.10.10->192.168.1.100 [55748]NAT: dyn flow info downl
```

```
<#root>
```

```
NAT-Device#
```

```
debug platform software nat all
```

```
NAT platform all events debugging is on
```

```
NAT-Device#
```

```
*May 18 23:52:20.301: FMANRP-NAT: Received flow data, action:
```

```
ADD <-- first packet in flow ADD operation
```

```
*May 18 23:52:20.301: id 5, flags 0x5, domain 0
```

```
src_local_addr 192.168.1.100, src_global_addr 172.16.10.10
```

```
, dst_local_addr 10.20.30.40,
```

```
<-- source translation
```

```
dst_global_addr 10.20.30.40,
```

```
src_local_port 52448, src_global_port 1024
```

```
,
```

```
<-- port translation
```

```
dst_local_port 23, dst_global_port 23,  
proto 6, table_id 0 inside_mapping_id 1,  
outside_mapping_id 0, inside_mapping_type 2,  
outside_mapping_type 0  
<snip>
```

## 하드웨어 확인

NAT Region 5(NAT 영역 5) 아래의 하드웨어에 NAT 규칙이 올바르게 설치되었는지 확인합니다.

```
<#root>
```

```
NAT-Device#
```

```
show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT
```

```
Printing entries for region
```

```
NAT_1
```

```
(370) type 6 asic 1
```

```
<-- NAT_1 empty due to no active flow
```

```
=====  
Printing entries for region NAT_2 (371) type 6 asic 1
```

```
=====  
Printing entries for region NAT_3 (372) type 6 asic 1
```

```
=====  
Printing entries for region NAT_4 (373) type 6 asic 1
```

```
=====  
Printing entries for region NAT_5 (374) type 6 asic 1
```

```
=====  
TAQ-2 Index-128 (A:0,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0  
Mask1 0300f000:00000000:00000000:00000000:00000000:00000000:ffffffc:00000000  
Key1 02009000:00000000:00000000:00000000:00000000:00000000:ac100a00:00000000  
AD 10087000:00000073
```

```
TAQ-2 Index-129 (A:0,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0  
Mask1 3300f000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:
```

```
ffffff00
```

```
Key1 21009000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:
```

```
c0a80100
```

```
AD 10087000:00000073
```

```
ffffff00 = 255.255.255.0 in hex for our subnet mask in NAT ACL
```

c0a80100 = 192.168.1.0 in hex for our network address in NAT ACL

마지막으로 플로우가 활성 상태일 때 NAT\_Region 1의 하드웨어 TCAM에 NAT 플로우가 프로그래밍되었는지 확인할 수 있습니다

<#root>

NAT-Device#

show ip nat translations

```
Pro Inside global      Inside local      Outside local  Outside global
tcp 172.16.10.10:1024  192.168.1.100:20027  10.20.30.40:23  10.20.30.40:23
```

NAT-Device#

show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT\_

Printing entries for region

NAT\_1

(370) type 6 asic 1

<-- NAT region 1

```
=====
TAQ-2 Index-32 (A:0,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0
Mask1 0000f000:ff00ffff:00000000:0000ffff:00000000:00000000:ffffffff:ffffffff
Key1 00009000:
```

06004e3b

:00000000:

00000017

:00000000:00000000:

0a141e28

:

c0a80164

AD 10087000:000000b0

```
TAQ-2 Index-33 (A:0,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Mask1 0000f000:ff00ffff:00000000:0000ffff:00000000:00000000:ffffffff:ffffffff
Key1 00009000:
```

06000017

:00000000:

00000400

:00000000:00000000:

0a141e28

:

0a141e28

AD 10087000:000000b1

Starting at Index-32 Key1 from right to left:

c0a80164

- 192.168.1.100 (inside local source address)

0a141e28

- 10.20.30.40 (inside global address/outside local address)

00000017

- 23 (TCP destination port)

06004e3b

- TCP source port 20027 (4e3b) and TCP protocol 6

Starting at Index-33 Key1 from right to left:

0a141e28

- 10.20.30.40 (outside global address/outside local address)

ac100a0a

- 172.16.10.10 (inside global)

00000400

- TCP inside global source port 1024

06000017

- TCP protocol 6 and TCP source port 23

## 패킷 레벨 디버그

하드웨어의 NAT 규칙과 일치하는 흐름의 첫 번째 패킷은 처리할 디바이스 CPU에 펀팅해야 합니다. . punt 경로 관련 디버그 출력을 보려면 FED punt path traces to debug level to ensure the packet is punt. CPU 리소스가 필요한 NAT 트래픽은 트랜짓 트래픽 CPU 대기열로 이동합니다.

트랜짓 트래픽 CPU 큐에 능동적으로 펀칭된 패킷이 표시되는지 확인합니다.

<#root>

NAT-DEVICE#

```
show platform software fed switch active punt cpuq clear <-- clear statistics
```

NAT-DEVICE#

```
show platform software fed switch active punt cpuq 18 <-- transit traffic queue
```

Punt CPU Q Statistics

=====

CPU Q Id :

18

CPU Q Name :

CPU\_Q\_TRANSIT\_TRAFFIC

Packets received from ASIC : 0

<-- no punt traffic for NAT

Send to IOSd total attempts : 0

Send to IOSd failed count : 0

RX suspend count : 0

RX unsuspend count : 0

RX unsuspend send count : 0

RX unsuspend send failed count : 0

RX consumed count : 0

RX dropped count : 0

RX non-active dropped count : 0

RX conversion failure dropped : 0

RX INTACK count : 0

RX packets dq'd after intack : 0

Active RxQ event : 0

RX spurious interrupt : 0

RX phy\_idb fetch failed: 0

RX table\_id fetch failed: 0

RX invalid punt cause: 0

Replenish Stats for all rxq:

-----

Number of replenish : 0

Number of replenish suspend : 0

Number of replenish un-suspend : 0

-----

NAT-DEVICE#

```
show platform software fed switch active punt cpuq 18 <-- after new translation
```

Punt CPU Q Statistics

=====

CPU Q Id : 18

CPU Q Name : CPU\_Q\_TRANSIT\_TRAFFIC

Packets received from ASIC : 5

<-- confirms the UADP ASIC punts to

```

Send to IOSd total attempts : 5
Send to IOSd failed count : 0
RX suspend count : 0
RX unsuspend count : 0
RX unsuspend send count : 0
RX unsuspend send failed count : 0
RX consumed count : 0
RX dropped count : 0
RX non-active dropped count : 0
RX conversion failure dropped : 0
RX INTACK count : 5
RX packets dq'd after intack : 0
Active RxQ event : 5
RX spurious interrupt : 0
RX phy_idb fetch failed: 0
RX table_id fetch failed: 0
RX invalid punt cause: 0

```

Replenish Stats for all rxq:


```

-----
Number of replenish : 18
Number of replenish suspend : 0
Number of replenish un-suspend : 0
-----

```

## NAT 확장 문제 해결

표에 나와 있는 최대 NAT TCAM 엔트리 수에 대한 현재 하드웨어 지원:

 참고: 각 활성 NAT 변환에는 2개의 TCAM 항목이 필요합니다.

플랫폼	최대 TCAM 항목 수
Catalyst 9300	5000
Catalyst 9400	14000
Catalyst 9500	14000
Catalyst 9500 고성능	15500
Catalyst 9600	15500

규모에 문제가 있는 것으로 의심되는 경우 플랫폼 제한에 대해 확인할 총 TCP/UDP NAT 변환 수를 확인할 수 있습니다.

```
<#root>
```

```
NAT-Device#
```

```
show ip nat translations | count tcp
```

```
Number of lines which match regexp =
```

```
621          <-- current number of TCP translations
```

```
NAT-Device#
```

```
show ip nat translations | count udp
```

```
Number of lines which match regexp =
```

```
4894          <-- current number of UDP translations
```

NAT TCAM 공간이 모두 소진된 경우 스위치 하드웨어의 NAT 모듈에서 이러한 변환을 처리할 수 없습니다. 이 시나리오에서 NAT 변환의 대상이 되는 트래픽은 처리할 디바이스 CPU에 대해 편팅 됩니다.

이로 인해 레이턴시가 발생할 수 있으며, NAT punt 트래픽을 담당하는 컨트롤 플레인 폴리서 큐에서 해당 값이 증가하는 삭제률 통해 확인할 수 있습니다. NAT 트래픽이 이동하는 CPU 대기열은 "전송 트래픽"입니다.

```
<#root>
```

```
NAT-Device#
```

```
show platform hardware fed switch active qos queue stats internal cpu policer
```

#### CPU Queue Statistics

```
=====
```

QId	PlcIdx	Queue Name	Enabled	(default) Rate	(set) Rate	Queue Drop(Bytes)	Queue Drop(Frames)
-----	--------	------------	---------	----------------	------------	-------------------	--------------------

```
-----
```

```
<snip>
```

14	13	Sw forwarding	Yes	1000	1000	0	0
15	8	Topology Control	Yes	13000	16000	0	0
16	12	Proto Snooping	Yes	2000	2000	0	0
17	6	DHCP Snooping	Yes	500	500	0	0
18	13	Transit Traffic	Yes	1000	1000	34387271	399507

```
<-- drops for NAT traffic headed towards the CPU
```

19	10	RPF Failed	Yes	250	250	0	0
20	15	MCAST END STATION	Yes	2000	2000	0	0

```
<snip>
```

17.x 코드에서 사용 가능한 NAT TCAM 공간을 확인합니다. 이 출력은 공간이 최대화되도록 NAT 템플릿이 활성화된 9300의 출력입니다.

```
<#root>
```

NAT-DEVICE#

show platform hardware fed switch 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
Mac Address Table	EM	I	32768	22	0.07%	0	0	0	22
Mac Address Table	TCAM	I	1024	21	2.05%	0	0	0	21
L3 Multicast	EM	I	8192	0	0.00%	0	0	0	0
L3 Multicast	TCAM	I	512	9	1.76%	3	6	0	0
L2 Multicast	EM	I	8192	0	0.00%	0	0	0	0
L2 Multicast	TCAM	I	512	11	2.15%	3	8	0	0
IP Route Table	EM	I	24576	16	0.07%	15	0	1	0
IP Route Table	TCAM	I	8192	25	0.31%	12	10	2	1
QOS ACL	TCAM	IO	1024	85	8.30%	28	38	0	19
Security ACL	TCAM	IO	5120	148	2.89%	27	76	0	45
Netflow ACL	TCAM	I	256	6	2.34%	2	2	0	2
<b>PBR ACL</b>	<b>TCAM</b>	<b>I</b>	<b>5120</b>	<b>24</b>	<b>0.47%</b>	<b>18</b>	<b>6</b>	<b>0</b>	<b>0</b>
Netflow ACL	TCAM	O	768	6	0.78%	2	2	0	2
Flow SPAN ACL	TCAM	IO	1024	13	1.27%	3	6	0	4
Control Plane	TCAM	I	512	281	54.88%	130	106	0	45
Tunnel Termination	TCAM	I	512	18	3.52%	8	10	0	0
Lisp Inst Mapping	TCAM	I	512	1	0.20%	0	0	0	1
Security Association	TCAM	I	256	4	1.56%	2	2	0	0
Security Association	TCAM	O	256	5	1.95%	0	0	0	5
CTS Cell Matrix/VPN Label	EM	O	8192	0	0.00%	0	0	0	0
CTS Cell Matrix/VPN Label	TCAM	O	512	1	0.20%	0	0	0	1
Client Table	EM	I	4096	0	0.00%	0	0	0	0
Client Table	TCAM	I	256	0	0.00%	0	0	0	0
Input Group LE	TCAM	I	1024	0	0.00%	0	0	0	0
Output Group LE	TCAM	O	1024	0	0.00%	0	0	0	0
Macsec SPD	TCAM	I	256	2	0.78%	0	0	0	2

16.x 코드에서 사용 가능한 NAT TCAM 공간을 확인합니다. 이 출력은 SDM Access 템플릿이 있는 9300에서 제공되므로 NAT TCAM 항목에 사용할 수 있는 공간이 최대화되지 않습니다.

<#root>

NAT-DEVICE#

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

CAM Utilization for ASIC [0]

Table	Max Values	Used Values
Unicast MAC addresses	32768/1024	20/21
L3 Multicast entries	8192/512	0/9
L2 Multicast entries	8192/512	0/11
Directly or indirectly connected routes	24576/8192	5/23
QoS Access Control Entries	5120	85



Security Access Control Entries	5120	145
Ingress Netflow ACEs	256	8
Policy Based Routing ACEs	1024	24 <-- NAT usage in PRB TCAM
Egress Netflow ACEs	768	8
Flow SPAN ACEs	1024	13
Control Plane Entries	512	255
Tunnels	512	17
Lisp Instance Mapping Entries	2048	3
Input Security Associations	256	4
SGT_DGT	8192/512	0/1
CLIENT_LE	4096/256	0/0
INPUT_GROUP_LE	1024	0
OUTPUT_GROUP_LE	1024	0
Macsec SPD	256	2

NAT를 선호하도록 SDM 템플릿을 변경하면 NAT TCAM에 사용 가능한 하드웨어 공간이 증가할 수 있습니다. 이렇게 하면 최대 TCAM 항목 수에 대한 하드웨어 지원이 향상됩니다.

<#root>

```
NAT-Device#conf t
Enter configuration commands, one per line. End with CNTL/Z.
NAT-Device(config)#
```

```
sdm prefer nat
```

NAT 템플릿으로 변환하기 전후의 SDM을 비교할 경우 사용 가능한 TCAM 공간이 QoS 액세스 제어 항목 및 PBR(Policy Based Routing) ACE에 대해 스왑되는지 확인할 수 있습니다.

PBR TCAM은 NAT가 프로그래밍되는 곳입니다.

<#root>

```
NAT-Device#
```

```
show sdm prefer
```

Showing SDM Template Info

```
This is the Access template.
Number of VLANs: 4094
Unicast MAC addresses: 32768
Overflow Unicast MAC addresses: 1024
L2 Multicast entries: 8192
Overflow L2 Multicast entries: 512
L3 Multicast entries: 8192
Overflow L3 Multicast entries: 512
Directly connected routes: 24576
Indirect routes: 8192
Security Access Control Entries: 5120
QoS Access Control Entries: 5120
```

```
Policy Based Routing ACEs: 1024 <-- NAT
```

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

```
NAT-Device#
```

```
show sdm prefer
```

```
Showing SDM Template Info
```

```
This is the NAT template.
```

```
Number of VLANs: 4094
```

```
Unicast MAC addresses: 32768
```

```
Overflow Unicast MAC addresses: 1024
```

```
L2 Multicast entries: 8192
```

```
Overflow L2 Multicast entries: 512
```

```
L3 Multicast entries: 8192
```

```
Overflow L3 Multicast entries: 512
```

```
Directly connected routes: 24576
```

```
Indirect routes: 8192
```

```
Security Access Control Entries: 5120
```

```
QoS Access Control Entries: 1024
```

```
Policy Based Routing ACEs: 5120 <-- NAT
```

```
<snip>
```

## AOT(Address Only Translation)

AOT는 NAT가 흐름의 레이어 4 포트가 아니라 IP 주소 필드만 변환하도록 요구될 때 사용할 수 있는 메커니즘입니다. 이 요구 사항이 충족되면 AOT는 하드웨어로 변환 및 전달할 플로우의 수를 크게 늘릴 수 있습니다.

- AOT는 대부분의 NAT 흐름이 단일 또는 소규모 대상 집합으로 향하는 경우 가장 효과적입니다.
- AOT는 기본적으로 비활성화되어 있습니다. 활성화된 후에는 현재 NAT 변환을 지워야 합니다.



참고: AOT는 PAT를 포함하지 않는 고정 NAT 및 동적 NAT에서만 지원됩니다.

---

즉, AOT를 허용하는 가능한 NAT 컨피그레이션은 다음과 같습니다.

```
#ip nat inside source static <source> <destination>  
#ip nat inside source list <list> pool <pool name>
```

다음 명령을 사용하여 AOT를 활성화할 수 있습니다.

<#root>

NAT-Device(config)#

no ip nat create flow-entries

AOT NAT 규칙이 올바르게 프로그래밍되었는지 확인합니다. 이 출력은 고정 NAT 변환의 결과입니다.

<#root>

NAT-DEVICE#

show running-config | include ip nat

ip nat outside

ip nat inside

no ip nat create flow-entries

<-- AOT enabled

ip nat inside source static 10.10.10.100 172.16.10.10

<-- static NAT enabled

NAT-DEVICE#

show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT

Printing entries for region NAT\_1 (376) type 6 asic 1

Printing entries for region NAT\_2 (377) type 6 asic 1

Printing entries for region NAT\_3 (378) type 6 asic 1

Printing entries for region NAT\_4 (379) type 6 asic 1

Printing entries for region NAT\_5 (380) type 6 asic 1

TAQ-1 Index-864 (A:0,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0

Mask1 3300f000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:ffffffffff

Key1 21009000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:

0a0a0a64

AD 10087000:00000073

TAQ-1 Index-865 (A:0,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0

Mask1 0300f000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:ffffffffff:00000000

Key1 02009000:00000000:00000000:00000000:00000000:00000000:00000000:00000000:

ac100a0a

:00000000

AD 10087000:00000073

```
0a0a0a64 = 10.10.10.100 (inside local)
ac100a0a = 172.16.10.10 (inside global)
```

흐름이 활성화될 때 소스 및 대상 IP 주소만 프로그래밍되었는지 확인하여 TCAM의 AOT 항목을 확인합니다.

```
<#root>
```

```
NAT-DEVICE#
```

```
show platform hardware fed switch active fwd-asic resource tcam table pbr record 0 format 0 | begin NAT
```

```
Printing entries for region NAT_1 (376) type 6 asic 1
```

```
Printing entries for region NAT_2 (377) type 6 asic 1
```

```
TAQ-1 Index-224 (A:0,C:1) Valid StartF-1 StartA-1 SkipF-0 SkipA-0
Mask1 0000f000:00000000:00000000:00000000:00000000:00000000:ffffff:ffffff
Key1 00009000:00000000:00000000:00000000:00000000:00000000:
```

```
c0a80164:0a0a0a64 <-- no L4 ports, only source and destination IP is programmed
```

```
AD 10087000:000000b2
```

```
TAQ-1 Index-225 (A:0,C:1) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Mask1 0000f000:00000000:00000000:00000000:00000000:00000000:ffffff:00000000
Key1 00009000:00000000:00000000:00000000:00000000:00000000:
```

```
ac100a0a
```

```
:00000000
```

```
AD 10087000:000000b3
```

```
0a0a0a64 = 10.10.10.100 in hex (inside local IP address)
```

```
c0a80164 = 192.168.1.100 in hex (outside local/outside global)
```

```
ac100a0a = 172.16.10.10 (inside global)
```

## 관련 정보

- [Catalyst 9300 17.3.x NAT 컨피그레이션 가이드](#)
- [Catalyst 9400 17.3.x NAT 컨피그레이션 가이드](#)
- [Catalyst 9500 17.3.x NAT 컨피그레이션 가이드](#)
- [Catalyst 9600 17.3.x NAT 컨피그레이션 가이드](#)
- [기술 지원 및 문서 - Cisco Systems](#)

Cisco 내부 정보

[CSCvz 46804](#) NAT TCAM 리소스가 소진되거나 NAT 항목을 성공적으로 프로그래밍할 수 없

을 때 syslog를 추가하는 기능이 향상되었습니다.

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.