

C8000v 라우터 성능 문제 해결

목차

[소개](#)

[사용되는 구성 요소](#)

[일반 문제 해결](#)

[오버런](#)

[기능 삭제](#)

[방울방울](#)

[하이퍼바이저](#)

[VMware ESXi](#)

[AWS](#)

[다중 TX 대기열](#)

[초과된 메트릭](#)

[Microsoft Azure](#)

[가속화된 네트워킹](#)

[Azure 및 단편화](#)

[Microsoft Azure에 대해 지원되는 인스턴스 유형](#)

[추가 리소스](#)

소개

이 문서에서는 퍼블릭 클라우드 및 ESXi 시나리오에서 C8000v 엔터프라이즈 라우터의 성능 문제를 해결하는 방법에 대해 설명합니다.

사용되는 구성 요소

이 문서의 정보는 다음 하드웨어 및 소프트웨어 구성 요소를 기반으로 합니다.

- 17.12 버전을 실행하는 C8000v
- ESXi 버전 7.0 U3

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.

일반 문제 해결

C8000v를 다른 환경에서 호스팅할 수 있지만, C8000v가 호스팅되는 위치와 상관없이 동일한 몇 가지 문제 해결 단계를 계속 진행할 수 있습니다. 기본 사항부터 시작하겠습니다. 가장 먼저 확인해야 할 사항은 디바이스가 용량 제한에 도달했는지 여부입니다. 이를 위해 다음 두 출력을 선택하여 시작할 수 있습니다.

1. show platform hardware qfp active datapath util summary - 이 명령은 C8000v가 모든 포트에서 수신하고 전송하는 입력/출력의 전체 정보를 제공합니다. 처리 로드 백분율에 관심을 집중해야 합니다. 100%에 도달하는 시나리오의 경우 용량 제한에 도달했음을 의미합니다

```
----- show platform hardware qfp active datapath utilization summary -----  
  
CPP 0:  
Input:   Total (pps)      5 secs      1 min      5 min      60 min  
         (bps)      93119      92938      65941      65131  
         (bps)      997875976  1000204000  708234904  699462016  
Output:  Total (pps)      93119      92949      65944      65131  
         (bps)      1052264704  1054733128  746744264  737395744  
Processing: Load (pct)      14         14         10         10
```

2 . show platform hardware qfp active datapath infrastructure sw-cio - 이 명령을 위의 명령의 좀 더 심층적인 버전으로 생각해 보십시오. QFP 사용률 번호에 포함되지 않은 IO 및 암호화 코어를 포함한 개별 코어에 대한 더 자세한 정보를 제공합니다. 이 기능은 특정 데이터 플레인 코어가 병목 현상을 일으키는 지 확인하려는 시나리오에서 매우 유용합니다.

9	Gi6	4:	2015	0	0	0	0	0	0	0	0	0	0
10	Gi7	4:	2002	0	0	0	0	0	0	0	0	0	0
11	vpg0	400:	490	0	0	0	0	0	0	0	0	0	0

Core Utilization over preceding 107352.2729 seconds

ID:	0	1	2	3	4	5	6	7	8	9	10	11
% PP:	2.98	2.01	1.81	1.67	1.60	1.53	1.35	1.30	1.25	1.19	2.19	1.19
% RX:	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
% TM:	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
% IDLE:	97.02	97.99	98.19	98.33	98.40	98.47	98.65	98.70	98.75	98.81	97.81	98.81

이제 플랫폼 제한에 도달했는지 여부를 결정했습니다. 다음 단계는 삭제를 확인하는 것입니다. 이는 기본적으로 성능 문제와 연결되어 있습니다. 발생 위치에 따라 고려할 수 있는 세 가지 유형의 드롭이 있습니다.

- 오버런: 이 유형의 패킷 삭제는 Rx 쪽에서 발생합니다. 이는 하나 이상의 코어의 처리 용량이 초과되었기 때문에 발생합니다.
- 기능 삭제: 이 유형의 패킷 삭제는 PPE에서 발생합니다. ACL 또는 QoS와 같은 라우터의 기능과 관련이 있습니다.
- Taildrops: 이 유형의 패킷 삭제는 Tx에서 발생합니다. Tx 버퍼의 혼잡으로 인해 발생합니다.

어떤 드롭이 발생하는지 확인하려면 다음 출력을 사용할 수 있습니다.

- show platform hardware qfp active drop state clear
- 인터페이스 표시
- show policy map interface

어떤 드롭에 직면하는지 식별하는 방법과 드롭을 완화하는 방법을 확인합니다. 그럼에도 불구하고 이 글에서는 Taildrops라고 하는 드롭이 가상 라우터에서 특히 문제를 해결하기 까다롭기 때문에 가장 큰 초점을 맞추고 있습니다.

오버런

Cisco IOS XE의 오버런 삭제는 네트워크 인터페이스에서 패킷을 처리하거나 버퍼에 저장할 수 있는 속도보다 빠른 속도로 수신할 때 발생합니다. 특히 인터페이스의 내부 버퍼(FIFO 큐)가 가득 차면 수신 데이터 속도가 하드웨어를 처리할 수 있는 능력을 초과하기 때문입니다. 그 결과, 새 수신 패킷을 저장할 수 없고 삭제되어 오버런 카운터가 증가합니다. 이는 기본적으로 인터페이스가 일시적으로 오버플로되어 발생한 패킷 손실입니다.

이 유형의 패킷 삭제는 Rx 쪽에서 발생합니다. 이는 하나 이상의 코어의 처리 용량이 초과되었고 Rx 스레드가 관련 PP 스레드에 수신 패킷을 배포할 수 없으며 인그레스 버퍼가 이미 가득 차기 때문에 발생합니다. 간단히 비유하자면, 계산대(인터페이스 하드웨어)가 제공할 수 있는 것보다 패킷이 더 빨리 도착하기 때문에 너무 가득 찬 계산대 대기열로 생각할 수 있습니다. 대기열이 가득 차면 신규 고객이 서비스를 받지 못한 채 떠나야 합니다. 이러한 상황은 초과 발생하는 손실입니다.

이 섹션에서는 하드웨어에 대해 설명하지만 C8000v는 소프트웨어 기반 라우터입니다. 이 경우 다음과 같은 이유로 오버런이 발생할 수 있습니다.

고(유휴 = 0%), 인터페이스 Gig2에 대한 낮은 I/O 크레딧을 기대할 수 있습니다.

```
#show platform hardware qfp active datapath infrastructure sw-cio
Credits Usage:
```

```
ID Port Wght Global WRKR0 WRKR1 WRKR2 Total
1 rc10 16: 487 0 0 25 512
1 rc10 32: 496 0 0 16 512
2 ipc 1: 490 0 0 21 511
3 vxe_punti 4: 459 0 0 53 512
4 Gi1 4: 477 0 0 35 512
5 Gi2 4: 474 0 0 38 512 <<< low/zero credits for interface Gig2:
```

```
Core Utilization over preceding 1.0047 seconds
```

```
-----
ID: 0 1 2
% PP: 0.77 0.00 0.00
% RX: 0.00 0.00 0.44
% TM: 0.00 0.00 5.63
% IDLE: 99.23 99.72 93.93 <<< the core ID relevant in this case would be PP#2
```

기능 삭제

패킷은 사용 가능한 모든 데이터 플레인 스레드에 의해 처리되며 소프트웨어 Rx 기능(x86) - LBD(로드 기반 배포)를 통해 QFP 코어의 가용성에 따라 엄격하게 배포됩니다. PPE에 도착하는 패킷은 특정 QFP 삭제 이유와 함께 삭제할 수 있으며, 이 출력은 다음과 같습니다.

```
#show drops
```

```
----- show platform hardware qfp active statistics drop detail -----
```

```
Last clearing of QFP drops statistics : never
```

ID	Global Drop Stats	Packets	Octets
319	BFDoffload	403	31434
139	Disabled	105	7487
61	Icmp	135	5994
94	Ipv4NoAdj	1	193
33	Ipv6NoRoute	2426	135856
215	UnconfiguredIpv4Fia	1937573	353562196
216	UnconfiguredIpv6Fia	8046173	1057866418

```
----- show platform hardware qfp active interface all statistics drop_summary -----
```

```
Drop Stats Summary:
```

```
note: 1) these drop stats are only updated when PAL
reads the interface stats.
```

```
2) the interface stats include the subinterface
```

```
Interface Rx Pkts Tx Pkts
```

GigabitEthernet1	9980371	0
GigabitEthernet2	4012	0

떨어지는 이유는 다양하고 보통 자명하다. 더 자세히 조사하기 위해 [패킷 추적을](#) 사용할 수 있습니다.

방울방울

앞서 언급했듯이 디바이스에서 패킷을 전송하려고 하지만 전송 버퍼가 꽉 찬 경우 taldrops가 발생합니다.

이 하위 섹션에서는 이러한 유형의 상황에 직면했을 때 검토할 수 있는 출력을 살펴볼 것입니다. 여기에서 볼 수 있는 가치는 무엇이며 문제를 완화하기 위해 무엇을 할 수 있는지 나타냅니다.

먼저, 여러분은 그들을 식별하는 방법을 알아야 합니다. 그러한 방법 중 하나는 쇼 인터페이스를 단순히 보는 것입니다. 다음과 같이 출력 감소가 증가하는 경우 주의하십시오.

```
GigabitEthernet2 is up, line protocol is up
Hardware is vNIC, address is 0050.56ad.c777 (bia 0050.56ad.c777)
Description: Connected-To-ASR Cloud Gateway
Internet address is 10.6.255.81/29
MTU 1500 bytes, BW 10000000 Kbit/sec, DLY 10 usec,
reliability 255/255, txload 2/255, rxload 3/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Full Duplex, 10000Mbps, link type is force-up, media type is Virtual
output flow-control is unsupported, input flow-control is unsupported
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:00, output 00:00:00, output hang never
Last clearing of "show interface" counters 03:16:21
Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 7982350 <<<<<<<<
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 150449000 bits/sec, 20461 packets/sec
5 minute output rate 89116000 bits/sec, 18976 packets/sec
```

이 명령은 특히 혼잡이 발생하는지 여부를 이해하는 데 유용합니다.

- show platform hardware qfp active datapath infrastructure - HQF는 'Hierarchical Queueing Framework'를 의미합니다. 이는 모듈형 QoS MQC(Command-Line Interface)를 사용하여 다양한 레벨(물리적, 논리적, 클래스)에서 QoS(Quality of Service) 관리를 가능하게 하는 기능입니다. 현재 RX 및 TX 비용을 보여줍니다. 출력에 표시된 대로 TX 대기열이 가득 차면(full 1959)

```
pmd b1689fc0 device Gi1
RX: pkts 5663120 bytes 1621226335 return 0 badlen 0
Out-of-credits: Hi 0 Lo 0
```

```
pkts/burst 1 cycl/pkt 1565 ext_cycl/pkt 1173
Total ring read 12112962299, empty 12107695202
TX: pkts 8047873582 bytes 11241140363740
pri-0: pkts 8047873582 bytes 11241140363740
pkts/send 3
Total: pkts/send 3 cycl/pkt 452
send 2013612969 sendnow 1810842
forced 2013274797 poll 724781 thd_poll 0
blocked 2197451 retries 7401 mbuf alloc err 0
TX Queue 0: full 1959 current index 0 hiwater 224
```

이 출력은 기본 하드웨어가 패킷 전송을 따라가지 못함을 나타냅니다. 기본 인터페이스를 디버깅하려면 C8000v 외부 및 C8000v가 실행 중인 기본 환경에서 기본 물리적 인터페이스에 보고된 추가 오류가 있는지 확인해야 합니다.

환경을 확인하기 위해 C8000v 라우터가 어떤 하이퍼바이저를 실행 중인지 확인하기 전에 수행할 수 있는 단계가 하나 있습니다. 이는 명령 show controller 출력을 확인하기 위한 것입니다. 그럼에도 불구하고 각 카운터가 무엇을 의미하는지, 어디에서 보아야 하는지에 대한 상실감을 발견할 수 있다.

먼저, 이 결과를 볼 때 염두에 두어야 할 중요한 세부 사항 중 하나는 정보가 대부분 vNIC 자체에서 제공된다는 것입니다. 각 NIC 드라이버에는 사용하는 특정 카운터 집합이 있습니다. 이러한 카운터는 드라이버별로 다를 수 있습니다. 다른 하이퍼바이저는 또한 제시된 것에 어떤 종류의 영향을 미칩니다. mbuf 카운터와 같은 일부 카운터는 DPDK 드라이버의 통계입니다. 이는 DPDK 드라이버에 따라 다를 수 있습니다. 실제 계산은 일반적으로 가상화 레이어의 하이퍼바이저에 의해 수행됩니다

```
GigabitEthernet2 - Gi2 is mapped to UIO on VXE
rx_good_packets 1590
tx_good_packets 1402515
rx_good_bytes 202860
tx_good_bytes 1857203911
rx_missed_errors 0
rx_errors 0
tx_errors 0
rx_mbuf_allocation_errors 0
rx_q0_packets 1590
rx_q0_bytes 202860
rx_q0_errors 0
tx_q0_packets 1402515
tx_q0_bytes 1857203911
rx_q0_drop_total 0
rx_q0_drop_err 0
rx_q0_drop_fcs 0
rx_q0_rx_buf_alloc_failure 0
tx_q0_drop_total 976999540797
tx_q0_drop_too_many_segs 0
tx_q0_drop_tso 0
tx_q0_tx_ring_full 30901211518
```

이 카운터를 해석하고 읽는 방법을 알아보려면 잠시 기다려 주십시오.

1. subX가 표시되면 하위 인터페이스임을 의미합니다. 이는 기본 인터페이스의 논리적 분할입니다. 일반적으로 sub0은 기본/기본 서브스크립션입니다. 여러 VLAN이 관련되어 있는 경우 이러한 VLAN이 자주 사용됩니다.
2. 그런 다음 "rx = receiving" 및 "tx = transmitting"이 있습니다.
3. 마지막으로 q0은 해당 인터페이스에서 사용되는 첫 번째/기본 대기열을 참조합니다

모든 카운터에 대한 설명은 없지만, 이 문서에서는 문제 해결에 관련될 수 있는 몇 가지 카운터에 대해 설명합니다.

- "RX_MISSED_ERRORS:"는 NIC 버퍼(Rx FIFO)가 오버플 상태가 될 때 표시됩니다. 이 조건은 삭제와 레이턴시의 증가로 이어집니다. 가능한 해결 방법은 NIC 버퍼를 늘리거나(여기서는 불가능) NIC 드라이버를 변경하는 것입니다.
- "tx_q0_drop_total" 및 "tx_q0_tx_ring_full": 호스트가 패킷을 삭제하고 있음을 알 수 있으며, 호스트가 C8000v를 역압박하므로 C8000v에서 Tail Drops가 발생합니다

위의 출력에는 "rx_missed_errors"가 표시되지 않습니다. 그러나, 우리가 테일 드롭에 초점을 맞추고 있기 때문에 우리는 "tx_q0_drop_total"과 "tx_q0_tx_ring_full"을 모두 볼 수 있습니다. 이를 통해 호스트의 기본 하드웨어로 인해 실제로 정체가 발생한다는 결론을 내릴 수 있습니다.

앞에서 언급한 것처럼, 각 하이퍼바이저는 제시된 항목에 어떤 종류의 영향을 미칩니다. 이 글에서는 C8000v를 호스팅할 수 있는 다양한 하이퍼바이저 간의 차이점을 살펴보면서 다음 섹션에서 이에 대해 중점적으로 설명합니다. 또한 이러한 유형의 문제를 각각의 문제에서 시도하고 완화하기 위한 다양한 권장 사항을 찾을 수 있습니다.

하이퍼바이저

하이퍼바이저는 CPU, 메모리, 스토리지 등의 하드웨어 리소스를 각 VM에 관리 및 할당하여 여러 운영 체제(가상 머신 또는 VM이라고 함)를 단일 물리적 하드웨어 호스트에서 실행할 수 있도록 하는 소프트웨어 레이어입니다. 이러한 가상 머신이 서로 간섭하지 않고 독립적으로 작동하도록 보장합니다.

Cisco Catalyst 8000V(C8000v)의 맥락에서 하이퍼바이저는 C8000v 가상 머신을 호스팅하는 플랫폼입니다. 어떤 하이퍼바이저가 C8000v를 호스팅하고 있는지 어떻게 알 수 있습니까? 우리에게 그 정보를 주는 다소 유용한 결과가 있다. 또한 Cisco 가상 라우터가 다음에 액세스할 수 있는 리소스의 종류도 확인할 수 있습니다.

```
C8000v#show platform software system all
Processor Details
=====
Number of Processors : 8
Processor : 1 - 8
vendor_id : GenuineIntel
cpu MHz : 2593.906
cache size : 36608 KB
Crypto Supported : Yes
model name : Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz
```

```
Memory Details
=====
```

Physical Memory : 32817356KB

VNIC Details

```
=====
Name Mac Address Driver Name Status Platform MTU
GigabitEthernet1 0022.480d.7a05 net_netvsc UP 1500
GigabitEthernet2 6045.bd69.83a0 net_netvsc UP 1500
GigabitEthernet3 6045.bd69.8042 net_netvsc UP 1500
```

Hypervisor Details

```
=====
Hypervisor: AZURE
Manufacturer: Microsoft Corporation
Product Name: Virtual Machine
Serial Number: 0000-0002-0201-5310-5478-4052-71
UUID: 8b06091c-f1d3-974c-85a5-a78dfb551bf2
Image Variant: None
```

VMware ESXi

ESXi는 VMware에서 개발한 Type-1 하이퍼바이저로서 물리적 서버에 직접 설치되어 가상화를 지원합니다. 하드웨어 리소스를 추상화하고 각 VM에 할당하여 여러 VM(가상 머신)을 하나의 물리적 서버에서 실행할 수 있습니다. C8000v 라우터는 이러한 VM 중 하나입니다.

혼잡이 발생하는 일반적인 시나리오를 살펴보는 것부터 시작할 수 있습니다. 이는 tx_q0_tx_ring_full 카운터를 확인하여 확인할 수 있습니다.

예:

```
----- show platform software vnic-if interface-mapping -----
```

```
-----
Interface Name Driver Name Mac Addr
-----
GigabitEthernet3 net_vmxnet3 <-- 0050.5606.2239
GigabitEthernet2 net_vmxnet3 0050.5606.2238
GigabitEthernet1 net_vmxnet3 0050.5606.2237
-----
```

```
GigabitEthernet3 - Gi3 is mapped to UIO on VXE
rx_good_packets 99850846
tx_good_packets 24276286
rx_good_bytes 78571263015
tx_good_bytes 14353154897
rx_missed_errors 0
rx_errors 0
tx_errors 0
rx_mbuf_allocation_errors 0
rx_q0packets 99850846
rx_q0bytes 78571263015
rx_q0errors 0
tx_q0packets 24276286
tx_q0bytes 14353154897
```

```
rx_q0_drop_total 0
rx_q0_drop_err 0
rx_q0_drop_fcs 0
rx_q0_rx_buf_alloc_failure 0
tx_q0_drop_total 160945155
tx_q0_drop_too_many_segs 0
tx_q0_drop_tso 0
tx_q0_tx_ring_full 5283588 <-----
```

이러한 혼잡은 C8000V가 VMXNET3 인터페이스를 통해 패킷을 전송하려고 시도할 때 발생합니다. 그러나 버퍼 링은 이미 패킷으로 가득 차 있으며, 결국 지연 또는 삭제로 끝납니다.

이러한 상황에서는 앞에서 언급한 대로 하이퍼바이저 측에서 이러한 드롭이 발생합니다. 모든 권장 사항이 충족되면 VMware 지원 팀에 문의하여 NIC에서 발생하는 상황을 파악하는 것이 좋습니다.

다음은 성능을 개선하는 방법에 대한 몇 가지 제안입니다.

- 최적의 성능을 위해 전용 vSwitch 및 업링크 사용
- 자체 물리적 업링크가 지원하는 전용 vSwitch에 C8000V를 할당함으로써 소음이 많은 인접 디바이스로부터 트래픽을 격리하고 공유 리소스 병목을 방지할 수 있습니다.

ESXi 측에서 살펴볼 만한 몇 가지 명령이 있습니다. 예를 들어 ESXi 인터페이스에서 패킷 손실을 확인하려면 다음을 수행합니다.

1. SSH를 활성화합니다.
2. SSH를 사용하여 ESXi에 연결합니다.
3. esxtop을 실행합니다.
4. n을 입력합니다.

esxtop 명령은 가상 머신의 네트워크 드라이버가 Rx 버퍼 메모리가 부족한 경우 가상 스위치에서 삭제된 패킷을 표시할 수 있습니다. esxtop에서는 가상 스위치에서 삭제된 패킷을 표시하지만 가상 스위치와 게스트 운영 체제 드라이버 사이에서 실제로 삭제됩니다.

%DRPTX 및 %DRPRX에서 삭제되는 패킷을 검색합니다.

```
12:34:43pm up 73 days 16:05, 907 worlds, 9 VMs, 53 vCPUs; CPU load average: 0.42, 0.42, 0.42
```

PORT-ID	USED-BY	TEAM	PNIC	DNAME	PKTTX/s	MbTX/s	PSZTX	PKTRX/s	MbRX/s	PSZRX	%DRPTX	%DRPRX
67108870	Management	n/a	vSwitch-to-9200		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
67108872	Shadow of vmnic1	n/a	vSwitch-to-9200		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
67108876	vmk1	vmnic1	vSwitch-to-9200		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
67108890	2101719:c8kv-gw-mgmt	vmnic1	vSwitch-to-9200		76724.83	792.35	1353.00	16180.39	9.30	75.00	0.00	0.00
100663305	Management	n/a	vSwitch-to-Cisc		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
100663307	Shadow of vmnic0	n/a	vSwitch-to-Cisc		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
100663309	vmk0	vmnic0	vSwitch-to-Cisc		3.64	0.01	280.00	3.29	0.00	80.00	0.00	0.00
100663310	2100707:gsoaresc-On_Prem	vmnic0	vSwitch-to-Cisc		0.00	0.00	0.00	2.43	0.00	60.00	0.00	0.00
100663311	2100993:cats-vmanage	void	vSwitch-to-Cisc		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
100663312	2100993:cats-vmanage	void	vSwitch-to-Cisc		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
100663313	2100993:cats-vmanage	vmnic0	vSwitch-to-Cisc		5.38	0.01	212.00	9.71	0.01	141.00	0.00	0.00
100663314	2101341:cats-vsmart	void	vSwitch-to-Cisc		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
100663315	2101341:cats-vsmart	vmnic0	vSwitch-to-Cisc		2.60	0.00	164.00	6.94	0.01	124.00	0.00	0.00

```

100663316 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00
100663317 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00
100663318 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 4.33 0.01 174.00 7.80 0.01 162.00 0.00 0.00
100663319 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 4.16 0.00 90.00 0.00 0.00
100663320 2101547:gdk-backup vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
100663321 2101703:sevvy vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
100663323 2101719:c8kv-gw-mgmt vmnic0 vSwitch-to-Cisc 16180.91 9.09 73.00 76755.87 792.44 1353.00 0.00
100663324 2137274:telemetry-server vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
100663335 2396721:netlab vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
2214592519 vmnic1 - vSwitch-to-9200 76727.26 792.38 1353.00 16182.64 9.30 75.00 0.00 0.00
2248146954 vmnic0 - vSwitch-to-Cisc 16189.05 9.32 75.00 76736.97 792.38 1353.00 0.00 0.00

```

이 명령은 호스트에 구성된 모든 NIC를 나열합니다.

```
esxcli network nic list
```

```
Name PCI Device Driver Admin Status Link Status Speed Duplex MAC Address MTU Description
```

```
-----
vmnic0 0000:01:00.0 igbn Up Up 1000 Full fc:99:47:49:c5:0a 1500 Intel(R) I350 Gigabit Network Connection
vmnic1 0000:01:00.1 igbn Up Up 1000 Full fc:99:47:49:c5:0b 1500 Intel(R) I350 Gigabit Network Connection
vmnic2 0000:03:00.0 ixgben Up Up 1000 Full a0:36:9f:1c:1f:cc 1500 Intel(R) Ethernet Controller 10 Gigabit Network Connection
vmnic3 0000:03:00.1 ixgben Up Up 1000 Full a0:36:9f:1c:1f:ce 1500 Intel(R) Ethernet Controller 10 Gigabit Network Connection

```

특정 VM에 할당된 vNIC의 상태를 확인하는 유용한 명령도 있습니다.

```
esxcli network vm list
```

```
World ID Name Num Ports Networks
```

```
-----
2137274 telemetry-server 1 Cisco Backbone 10.50.25.0/24
2101703 sevvy 1 Cisco Backbone 10.50.25.0/24
2396721 netlab 1 Cisco Backbone 10.50.25.0/24
2101547 gdk-backup 1 Cisco Backbone 10.50.25.0/24
2101522 cats-vbond 4 VPNO, VPNO, VPNO, VPNO
2101719 c8kv-gw-mgmt 2 c8kv-to-92001, c8kv-to-cisco
2100707 gsoaresc-On_Prem 1 Cisco Backbone 10.50.25.0/24
2100993 cats-vmanage 3 VPNO, VPNO, VPNO
2101341 cats-vsmart 2 VPNO, VPNO
[root@localhost:~]
```

C8kv-gw-mgmt(C8000v VM)를 보면 2개의 네트워크가 할당되어 있습니다.

- c8kv-92001
- c8kv-cisco

World ID를 사용하여 이 VM에 대한 자세한 정보를 찾을 수 있습니다.

```
[root@localhost:~] esxcli network vm port list -w 2101719
Port ID: 67108890
vSwitch: vSwitch-to-9200L
Portgroup: c8kv-to-92001
DVPort ID:
MAC Address: 00:0c:29:31:a6:b6
IP Address: 0.0.0.0
Team Uplink: vmnic1
Uplink Port ID: 2214592519
Active Filters:

Port ID: 100663323
vSwitch: vSwitch-to-Cisco
Portgroup: c8kv-to-cisco
DVPort ID:
MAC Address: 00:0c:29:31:a6:ac
IP Address: 0.0.0.0
Team Uplink: vmnic0 <----
Uplink Port ID: 2248146954
Active Filters:
[root@localhost:~]
```

이 정보가 있으면 vSwitch가 어떤 네트워크에 할당되었는지 식별할 수 있습니다.

vSwitch에 할당된 물리적 NIC의 일부 트래픽 통계를 확인하려면 다음 명령을 사용합니다.

```
# esxcli network nic stats get -n <vmnic>
```

이 명령은 수신된 패킷, 수신된 바이트, 삭제된 패킷 및 수신된 오류 등의 정보를 표시합니다. 이렇게 하면 NIC에서 드롭이 발생하는지 식별하는 데 도움이 될 수 있습니다.

```
[root@localhost:~] esxcli network nic stats get -n vmnic0
NIC statistics for vmnic0
Packets received: 266984237
Packets sent: 123640666
Bytes received: 166544114308
Bytes sent: 30940114661
Receive packets dropped: 0
Transmit packets dropped: 0
Multicast packets received: 16773454
Broadcast packets received: 36251726
Multicast packets sent: 221108
Broadcast packets sent: 1947649
Total receive errors: 0
Receive length errors: 0
Receive over errors: 0
Receive CRC errors: 0
Receive frame errors: 0
Receive FIFO errors: 0
Receive missed errors: 0
Total transmit errors: 0
Transmit aborted errors: 0
Transmit carrier errors: 0
```

Transmit FIFO errors: 0
Transmit heartbeat errors: 0
Transmit window errors: 0

호스트 및 가상 머신의 설정을 수정하여 ESXi 환경에서 실행되는 Cisco Catalyst 8000V의 성능을 향상시킬 수 있는 몇 가지 구성을 확인할 수 있습니다.

- 가상 하드웨어를 설정합니다. CPU 예약 설정을 최대값으로 설정합니다.
- 가상 하드웨어에서 모든 게스트 메모리 예약: 메모리.
- 가상 하드웨어에서 VMware Paravirtual을 선택합니다. SCSI 컨트롤러.
- 가상 하드웨어에서: 네트워크 어댑터: 어댑터 유형 옵션에서 지원되는 NIC에 대해 SR-IOV를 선택합니다.
- General Guest OS Version(일반 게스트 OS 버전) > VM Options(VM 옵션) 옵션을 Other 3.x or later Linux (64-bit)(기타 3.x 이상 Linux(64비트))로 설정합니다.
- Advanced Latency Sensitivity(고급 레이턴시 감도) 아래의 VM Options(VM 옵션) 옵션을 High(높음)로 설정합니다.
- VM Options(VM 옵션) > Advanced Edit Configuration(고급 수정 컨피그레이션) 아래에서 SRIOV NIC와 동일한 NUMA 노드에 "numa.nodeAffinity"를 추가합니다
- 하이퍼바이저 성능 설정을 활성화합니다.
- 지원되는 물리적 NIC에서 SR-IOV를 활성화하여 vSwitch의 오버헤드를 제한합니다.
- VM의 vCPU가 물리적 NIC와 동일한 NUMA 노드에서 실행되도록 구성합니다.
- VM Latency Sensitivity(VM 레이턴시 감도)를 High(높음)로 설정합니다.

AWS

C8000v는 Amazon Virtual Private Cloud(VPC) 내에서 Amazon Machine Image(AMI)로 시작함으로써 AWS에서의 배포를 지원하며, 사용자가 네트워크 리소스를 위해 논리적으로 격리된 AWS 클라우드 섹션을 프로비저닝할 수 있도록 합니다.

다중 TX 대기열

AWS에서 실행되는 C8000v의 주요 기능은 Multi-TX Queue(Multi-TXQ)를 사용하는 것입니다. 이러한 대기열은 내부 처리 오버헤드를 줄이고 확장성을 향상하는 데 도움이 됩니다. 큐가 여러 개이면 수신 및 발신 패킷을 올바른 vCPU(가상 CPU)에 더 빠르고 간단하게 할당할 수 있습니다.

RX/TX 큐가 vCPU별로 할당되는 일부 시스템과 달리 C8000v에서는 이러한 큐가 인터페이스별로 할당됩니다. RX(수신) 및 TX(전송) 큐는 Catalyst 8000V 애플리케이션과 AWS 인프라 또는 하드웨어 간의 연결 지점 역할을 하며, 네트워크 트래픽의 송수신 방식을 관리합니다. AWS는 인스턴스 유형에 따라 각 인터페이스에 사용할 수 있는 RX/TX 큐의 수와 속도를 제어합니다.

여러 TX 큐를 만들려면 Catalyst 8000V에 여러 인터페이스가 있어야 합니다. 여러 TX 대기열이 활성화된 경우 디바이스는 흐름의 5-튜플(소스 IP, 목적지 IP, 소스 포트, 목적지 포트, 프로토콜)을 기반으로 해싱 방법을 사용하여 패킷 흐름의 순서를 유지합니다. 이 해싱은 각 플로우에 사용할 TX 대

기열을 결정합니다.

사용자는 AWS 인스턴스에 연결된 동일한 물리적 NIC(Network Interface Card)를 사용하여 Catalyst 8000V에서 여러 인터페이스를 생성할 수 있습니다. 이 작업은 루프백 인터페이스를 구성하거나 보조 IP 주소를 추가하여 수행합니다.

Multi-TXQ의 경우 발신 트래픽을 처리하기 위한 여러 전송 큐가 있습니다. 이 예에는 12개의 TX 대기열(0~11개의 번호)이 있습니다. 이 설정을 사용하면 각 대기열을 개별적으로 모니터링하여 대기열이 가득 차는지 확인할 수 있습니다.

출력을 보면 TX Queue 8에 매우 높은 "full" 카운터(56,406,998)가 있으며, 이는 버퍼가 자주 채워진다는 것을 의미합니다. 다른 TX 대기열은 "전체" 카운터에 대해 0을 표시하여 혼잡하지 않음을 나타냅니다.

```
Router#show platform hardware qfp active datapath infrastructure sw-cio
pmd b17a2f00 device Gi2
RX: pkts 9525 bytes 1229599 return 0 badlen 0
Out-of-credits: Hi 0 Lo 0
pkts/burst 1 cycl/pkt 560 ext_cycl/pkt 360
Total ring read 117322273, empty 117312792
TX: pkts 175116324 bytes 246208197526
pri-0: pkts 157 bytes 10238
pkts/send 1
pri-1: pkts 75 bytes 4117
pkts/send 1
pri-2: pkts 91 bytes 6955
pkts/send 1
pri-3: pkts 95 bytes 8021
pkts/send 1
pri-4: pkts 54 bytes 2902
pkts/send 1
pri-5: pkts 75 bytes 4082
pkts/send 1
pri-6: pkts 104 bytes 8571
pkts/send 1
pri-7: pkts 74 bytes 4341
pkts/send 1
pri-8: pkts 175115328 bytes 246208130411
pkts/send 2
pri-9: pkts 85 bytes 7649
pkts/send 1
pri-10: pkts 106 bytes 5784
pkts/send 1
pri-11: pkts 82 bytes 7267
pkts/send 1
Total: pkts/send 2 cycl/pkt 203
send 68548581 sendnow 175024880
forced 1039215617 poll 1155226129 thd_poll 0
blocked 2300918060 retries 68534370 mbuf alloc err 0
TX Queue 0: full 0 current index 0 hiwater 0
TX Queue 1: full 0 current index 0 hiwater 0
TX Queue 2: full 0 current index 0 hiwater 0
TX Queue 3: full 0 current index 0 hiwater 0
TX Queue 4: full 0 current index 0 hiwater 0
TX Queue 5: full 0 current index 0 hiwater 0
```

```
TX Queue 6: full 0 current index 0 hiwater 0
TX Queue 7: full 0 current index 0 hiwater 0
TX Queue 8: full 56406998 current index 224 hiwater 224 <<<<<<<<<<<<
TX Queue 9: full 0 current index 0 hiwater 0
TX Queue 10: full 0 current index 0 hiwater 0
TX Queue 11: full 0 current index 0 hiwater 0
```

TX 대기열의 "전체" 카운터를 모니터링하면 전송 대기열이 오버로드되었는지 식별할 수 있습니다. 특정 TX 대기열에서 지속적으로 증가하는 "전체" 카운트는 디바이스에 부담을 주는 트래픽 흐름을 가리킵니다. 이를 해결하려면 트래픽 밸런싱, 컨피그레이션 조정 또는 성능 향상을 위한 리소스 확장을 수행해야 합니다.

초과된 메트릭

AWS는 서로 다른 인스턴스 크기에서 일관되고 고품질의 네트워크 성능을 보장하기 위해 인스턴스 수준에서 특정 네트워크 제한을 설정합니다. 이러한 제한은 모든 사용자의 안정적인 네트워킹을 유지하는 데 도움이 됩니다.

디바이스에서 `show controllers` 명령을 사용하여 이러한 제한 및 관련 통계를 확인할 수 있습니다. 출력에는 많은 카운터가 포함되지만 여기서는 네트워크 성능 모니터링에 가장 중요한 카운터에만 중점을 둡니다.

```
c8kv-2#sh control | inc exceed
<snipped>
bw_in_allowance_exceeded 0
bw_out_allowance_exceeded 0
pps_allowance_exceeded 0
contrack_allowance_exceeded 0
linklocal_allowance_exceeded 0
<snipped>
```

이제 해당 카운터가 정확히 무엇을 가리키는지 살펴볼 수 있습니다.

- `bw_in_allowance_exceeded`: 수신 대역폭이 인스턴스 제한을 초과하여 대기 중이거나 삭제된 패킷 수입니다.
- `bw_out_allowance_exceeded`: 발신 대역폭이 인스턴스 제한을 초과했기 때문에 대기 중이거나 삭제된 패킷 수입니다.
- `pps_allowance_exceeded`: PPS(total packets per second)가 인스턴스 제한을 초과했기 때문에 대기열에 있거나 삭제된 패킷 수입니다.
- `contrack_allowance_exceeded`: 인스턴스 유형에 허용된 최대값에 도달한 추적된 연결 수입니다.
- `linklocal_allowance_exceeded`: 로컬 프록시 서비스(예: Amazon DNS, Instance Metadata Service 및 Time Sync Service)에 대한 트래픽이 네트워크 인터페이스에 대한 PPS 제한을 초과하여 삭제된 패킷 수입니다. 이는 맞춤형 DNS 리졸버에는 영향을 주지 않습니다.

C8000v 성능에 미치는 영향:

- 이러한 카운터가 증가하고 성능 문제가 발생할 경우 C8000v 라우터가 항상 문제가 되는 것은 아닙니다. 대신 사용 중인 AWS 인스턴스가 용량 제한에 도달했음을 나타내는 경우가 많습니다. AWS 인스턴스의 사양을 확인하여 트래픽 요구 사항을 처리할 수 있는지 확인할 수 있습니다.

Microsoft Azure

이 섹션에서는 Microsoft Azure와 Cisco C8000v 가상 라우터가 결합하여 클라우드에서 확장 가능하고 안전한 고성능 가상 네트워킹 솔루션을 제공하는 방법을 살펴봅니다.

AN(Accelerated Networking) 및 패킷 단편화가 성능에 어떤 영향을 미칠 수 있는지 살펴봅니다. Microsoft Azure에 지원되는 인스턴스 유형을 사용하는 것의 중요성 검토

가속화된 네트워킹

C8000v가 Microsoft Azure 클라우드에서 호스팅되는 성능 문제 간과할 수 없는 한 가지 측면은 Accelerated Network의 활성화 여부입니다. 라우터의 성능을 크게 향상시킵니다. 간단히 말해, 가속화된 네트워킹은 Cisco Catalyst 8000V VM과 같은 VM에서 단일 루트 I/O 가상화(SR-IOV)를 지원합니다. 가속화된 네트워킹 경로는 가상 스위치를 우회하고 네트워크 트래픽 속도를 높이며 네트워킹 성능을 개선하고 네트워크 지연 및 지터를 줄입니다.

Accelerated Network가 활성화되어 있는지 확인하는 매우 간단한 방법이 있습니다. 즉 show controllers 출력을 확인하고 특정 카운터가 있는지 여부를 확인합니다.

```
----- show controllers -----
```

```
GigabitEthernet1 - Gi1 is mapped to UIO on VXE
rx_good_packets 6497723453
tx_good_packets 14690462024
rx_good_bytes 2271904425498
tx_good_bytes 6276731371987
```

```
rx_q0_good_packets 58576251
rx_q0_good_bytes 44254667162
```

```
vf_rx_good_packets 6439147188
vf_tx_good_packets 14690462024
vf_rx_good_bytes 2227649747816
vf_tx_good_bytes 6276731371987
```

찾고 있는 카운터는 `vf_rx_good_packets`와 같은 `vf`로 시작하는 카운터입니다. 이러한 카운터가 있음을 확인하면 가속 네트워킹이 활성화되어 있는지 확인할 수 있습니다.

Azure 및 단편화

단편화는 성능에 부정적인 영향을 미칠 수 있습니다. 성능에 미치는 영향의 주요 원인 중 하나는 패킷의 프래그먼트화 및 재결합에 따른 CPU/메모리 효과입니다. 네트워크 디바이스가 패킷을 프래그먼트화해야 하는 경우 프래그먼트화를 수행하기 위해 CPU/메모리 리소스를 할당해야 합니다.

패킷이 리어셈블될 때도 같은 현상이 발생합니다. 네트워크 디바이스는 프래그먼트를 받을 때까지 모두 저장해야 합니다. 그러면 프래그먼트를 원래 패킷으로 리어셈블할 수 있습니다.

Azure는 가속화된 네트워킹으로 조각화된 패킷을 처리하지 않습니다. VM이 프래그먼트된 패킷을 수신하면 비가속 경로가 이를 처리합니다. 따라서 프래그먼트화된 패킷은 낮은 레이턴시, 줄어든 지터, 높은 초당 패킷 수 등 가속화된 네트워킹의 이점을 놓칩니다. 이 때문에 가능하면 단편화를 피하는 것이 좋습니다.

기본적으로 Azure는 VM에 도착한 조각화된 패킷을 삭제합니다. 즉, 패킷이 소스 엔드포인트의 전송 시퀀스와 일치하지 않습니다. 이 문제는 패킷이 인터넷 또는 기타 대형 WAN을 통해 이동할 때 발생할 수 있습니다.

Microsoft Azure에 대해 지원되는 인스턴스 유형

C8000v는 Cisco 표준에 따라 지원되는 인스턴스 유형을 사용해야 합니다. [Cisco Catalyst 8000V Edge Software Installation And Configuration Guide](#)에서 확인할 수 있습니다.

그 이유는 해당 목록의 인스턴스 유형이 C8KV가 제대로 테스트된 인스턴스이기 때문입니다. 이제 C8000v가 나열되지 않은 인스턴스 유형에서 작동하는 경우 유효한 질문이 있습니까? 답은 아마도 '그렇다'일 것이다. 그러나 성능 문제처럼 복잡한 문제를 해결할 때는 알려지지 않은 또 다른 요소가 이 문제에 추가하지 않습니다. 이러한 이유만으로 Cisco TAC에서는 항상 지원되는 인스턴스 유형을 유지할 것을 권장합니다.

추가 리소스

성능 문제는 현재 발생한 경우에만 진정으로 해결할 수 있습니다. 그러나, 이것은 어떤 주어진 순간에 일어날 수 있기 때문에 잡아내기 어려울 수 있습니다. 따라서 이 EEM 스크립트를 제공합니다. 패킷이 삭제되기 시작하고 성능 문제가 발생하는 순간 중요한 출력을 캡처하는 데 도움이 됩니다.

```
ip access-list extended TAC
permit ip host host
```

```
permit ip host
```

host

conf t

```
event manager applet CONNECTIONLOST1 authorization bypass
event track 100 state down maxrun 500
action 0010 syslog msg "Logging information to file bootflash:SLA-DROPS.txt and bootflash:FIASLA_Decode.txt"
action 0020 cli command "enable"
action 0021 cli command "term length 0"
action 0022 cli command "term exec prompt timestamp"
action 0023 cli command "term exec prompt expand"
action 0095 cli command "show clock | append bootflash:SLA-DROPS.txt"
action 0096 cli command "show platform hardware qfp active statistics drop detail | append bootflash:SLA-DROPS.txt"
action 0097 cli command "show logging | append bootflash:SLA-DROPS.txt"
action 0099 cli command "show interfaces summary | append bootflash:SLA-DROPS.txt"
action 0100 cli command "show interfaces | append bootflash:SLA-DROPS.txt"
action 0101 cli command "show platform hardware qfp active statistics drop clear"
action 0102 cli command "debug platform packet-trace packet 2048 fia-trace"
action 0103 cli command "debug platform packet-trace copy packet both"
action 0104 cli command "debug platform condition ipv4 access-list TAC both"
action 0105 cli command "debug platform condition start"
action 0106 cli command "show platform hardware qfp active data infrastructure sw-cio | append bootflash:SLA-DROPS.txt"
action 0110 wait 60
action 0111 cli command "debug platform condition stop"
action 0112 cli command "show platform packet-trace packet all decode | append bootflash:FIASLA_Decode.txt"
action 0120 cli command "show platform packet-trace statistics | append bootflash:FIASLA_Decode.txt"
action 0121 cli command "show platform packet-trace summary | append bootflash:FIASLA_Decode.txt"
action 0122 cli command "show platform hardware qfp active datapath utilization summary | append bootflash:SLA-DROPS.txt"
action 0123 cli command "show platform hardware qfp active statistics drop detail | append bootflash:SLA-DROPS.txt"
action 0124 cli command "show platform hardware qfp active infrastructure bqs queue output default all | append bootflash:SLA-DROPS.txt"
action 0125 cli command "show platform software status control-processor brief | append bootflash:SLA-DROPS.txt"
action 0126 cli command "show platform hardware qfp active datapath infrastructure sw-pktmem | append bootflash:SLA-DROPS.txt"
action 0127 cli command "show platform hardware qfp active infrastructure punt statistics type per-cause | append bootflash:SLA-DROPS.txt"
action 0128 cli command "show platform hardware qfp active statistics drop | append bootflash:SLA-DROPS.txt"
action 0129 cli command "show platform hardware qfp active infrastructure bqs queue output default all | append bootflash:SLA-DROPS.txt"
action 0130 cli command "show platform hardware qfp active data infrastructure sw-hqf config 0 0 | append bootflash:SLA-DROPS.txt"
action 0131 cli command "show platform hardware qfp active feature lic-bw oversubscription | append bootflash:SLA-DROPS.txt"
action 0132 cli command "show platform hardware qfp active data infrastructure sw-hqf config 0 0 | append bootflash:SLA-DROPS.txt"
action 0133 cli command "show platform hardware qfp active data infrastructure sw-cio | append bootflash:SLA-DROPS.txt"
action 0134 cli command "show platform hardware qfp active data infrastructure sw-hqf sched | append bootflash:SLA-DROPS.txt"
action 0135 cli command "show platform hardware qfp active data infrastructure sw-dist | append bootflash:SLA-DROPS.txt"
action 0136 cli command "show platform hardware qfp active data infrastructure sw-nic | append bootflash:SLA-DROPS.txt"
action 0137 cli command "show platform hardware qfp active data infrastructure sw-pktmem | append bootflash:SLA-DROPS.txt"
action 0138 cli command "show controllers | append bootflash:SLA-DROPS.txt"
action 0139 cli command "show platform hardware qfp active datapath pmd controllers | append bootflash:SLA-DROPS.txt"
action 0140 cli command "show platform hardware qfp active datapath pmd system | append bootflash:SLA-DROPS.txt"
action 0141 cli command "show platform hardware qfp active datapath pmd static-if-config | append bootflash:SLA-DROPS.txt"
action 0150 cli command "clear platform condition all"
action 0151 cli command "clear platform packet-trace statistics"
action 0152 cli command "clear platform packet-trace configuration"
action 0153 cli command "show log | append bootflash:throughput_levelinfoSLA.txt"
action 0154 cli command "show version | append bootflash:throughput_levelinfoSLA.txt"
action 0155 cli command "show platform software system all | append bootflash:throughput_levelinfoSLA.txt"
action 0156 syslog msg "EEM script and FIA trace completed."
action 0180 cli command "conf t"
action 0181 cli command "no event manager applet CONNECTIONLOST1"
end
```


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.