엔터프라이즈 네트워크에서 라우터 문제 해결

목차

소개

사전 요구 사항

- 요구 사항
- 사용되는 구성 요소

배경 정보

레이턴시 정의

레이턴시 사용량

대기 시간 문제에 접근

일반적인 원인 트러블슈팅

- 플랫폼 관련
  - 높은 CPU
- 트래픽 관련
  - MTU 및 단편화
- 설계 관련
  - 차선의 경로
  - QoS(Quality of Service)

기타 성능 문제

- 삭제
- TCP 재전송
- 초과 유입 및 병목 현상

관련 정보

소개

이 문서에서는 Cisco 라우터를 사용하여 엔터프라이즈 네트워크의 지연 문제를 식별, 트러블슈팅 및 해결하는 방법을 설명합니다.

사전 요구 사항

요구 사항

이 문서에 대한 특정 전제 조건 또는 요구 사항이 없습니다.

사용되는 구성 요소

이 문서는 특정 소프트웨어 버전 및 하드웨어 유형에 제한되지 않지만, 명령은 ASR 1000, ISR 4000 및 Catalyst 8000 제품군과 같은 Cisco IOS® XE 라우터에 적용할 수 있습니다.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바

이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 정보

이 문서에서는 일반적인 대기 시간 문제를 이해, 격리 및 트러블슈팅하는 기본 설명서에 대해 설명하고, 근본 원인 및 모범 사례를 탐지하는 데 유용한 명령/디버그를 제공합니다. 모든 가능한 변수와 시나리오는 고려될 수 없으며 심층 분석은 특정 상황에 따라 달라집니다.

레이턴시 정의

일반적으로 저장 및 전달 디바이스에 대한 엄격한 정의(RFC 1242)의 견적을 작성할 때 레이턴시는 입력 프레임의 마지막 비트가 입력 포트에 도달하는 시점부터 시작하여 출력 프레임의 첫 번째 비트가 출력 포트에 나타나는 시점까지의 시간 간격입니다.

네트워크 레이턴시는 단순히 네트워크 전반의 데이터 전송 지연을 의미할 수 있습니다. 현실적인 문제에서 이 정의는 시작점에 불과합니다. 모든 특정 사례에서 말하는 지연 문제를 정의해야 합니다. 분명한 것처럼 보이지만 문제를 해결하기 위해 필요한 첫 번째 단계는 문제를 정의하는 것입니다.

레이턴시 사용량

많은 애플리케이션이 실시간 커뮤니케이션 및 비즈니스 운영에 짧은 레이턴시를 요구합니다. 하드웨어 및 소프트웨어가 매일 개선됨에 따라 미션 크리티컬 연산, 온라인 회의 애플리케이션, 스트리밍 등에 더 많은 애플리케이션을 사용할 수 있습니다. 이와 같은 방식으로 네트워크 트래픽이 계속 증가하고 최적화된 네트워크 설계 및 더 나은 장치 성능에 대한 요구도 증가하고 있습니다.

더 우수한 사용자 경험을 제공하고 레이턴시에 민감한 애플리케이션에 필요한 최소 요구 사항을 제공하는 것 외에도, 네트워크의 레이턴시 문제를 효과적으로 식별하고 줄이면 네트워크에서 많은 시간과 리소스를 크게 절약할 수 있습니다.

대기 시간 문제에 접근

이러한 유형의 문제에서 가장 어려운 부분은 고려해야 하는 변수의 수이며 단 하나의 실패 지점이 있을 수 없습니다. 따라서 레이턴시를 정의하는 것은 이를 해결하는 중요한 열쇠가 되며, 유용한 문제 설명을 위해 고려해야 하는 몇 가지 측면이 그 다음입니다.

1. 기대 및 탐지

원하는 레이턴시, 예상 또는 베이스라인 작업 레이턴시, 현재 레이턴시를 구분하는 것이 중요합니다. 설계, 네트워크 상의 공급자 또는 장치에 따라 원하는 레이턴시를 달성할 수 없는 경우도 있으므로 정상 조건에서 실제 레이턴시를 측정하는 것이 좋지만, 잘못된 숫자를 피하기 위해 측정 방법에 일관성을 유지해야 합니다. IP SLA 및 네트워크 분석기 툴을 사용하면 이러한 문제를 해결할 수 있습니다.

애플리케이션 또는 심지어 IP SLA별로 레이턴시를 식별하는 데 가장 많이 사용되는 기본 툴 중 하

나는 ICMP 또는 ping을 통한 것입니다.

<#root>

```
Router#

ping

 198.51.100.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 198.51.100.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5),

round-trip min/avg/max

 =

2/109/541 ms
```

ping은 연결 여부 확인 외에도 소스에서 대상까지의 RTT(왕복 시간), 즉 최소(2), 평균(109) 및 최대(541)를 밀리초 단위로 알려줍니다. 즉, 라우터가 요청을 전송한 시점부터 디바이스 목적지에서 응답을 수신하는 시점까지의 기간입니다. 그러나 얼마나 많은 홉이나 더 깊은 정보를 보여주지는 않지만, 쉽고 빠르게 문제를 탐지할 수 있는 방법입니다.

2. 격리

Traceroute는 ping과 마찬가지로 격리를 위한 시작점으로 사용할 수 있으며 홉과 홉당 RTT를 검색합니다.

<#root>

```
Router#

traceroute

 198.51.100.1
Type escape sequence to abort.
Tracing the route to 198.51.100.1
VRF info: (vrf in name/id, vrf out name/id)
  1 10.0.3.1 5 msec 6 msec 1 msec
  2 10.0.1.1 1 msec 1 msec 1 msec
  3 10.60.60.1 1 msec 1 msec 1 msec
  4 10.90.0.2

362 msec 362 msec 362 msec



<<<< you can see the RTT of the three probes only on both hops

  5 10.90.1.2

363 msec 363 msec 183 msec


  6 10.90.7.7 3 msec 2 msec 2 msec
```

Traceroute는 TTL(TimeTo Live)이 1인 패킷을 전송하여 작동합니다. 첫 번째 홉은 TTL이 만료되어 RTT가 측정되었으므로 패킷을 전달할 수 없음을 나타내는 ICMP 오류 메시지를 다시 보냅니다. 그런 다음 두 번째 패킷이 TTL이 2로 재전송되며 두 번째 홉은 TTL이 만료되었음을 반환합니다. 이 프로세스는 목적지에 도달할 때까지 계속됩니다.

예를 들어, 특정 호스트 2개로 범위를 좁힐 수 있으며 격리된 상태에서 시작할 수 있습니다.

이러한 명령은 문제를 쉽게 식별할 수 있는 유용한 명령임에도 불구하고 프로토콜, 패킷 표시 및 크기(두 번째 단계로 설정할 수 있지만), 여러 가지 요인 중 서로 다른 IP 소스, 대상 등의 다른 변수를 고려하지 않습니다.

레이턴시는 매우 광범위한 개념일 수 있으며 애플리케이션, 검색, 통화 또는 특정 작업에서만 증상을 자주 볼 수 있습니다. 제한해야 할 첫 번째 사항 중 하나는 영향을 이해하고 문제를 더 자세히 정의하며, 다음 질문에 답하고 요소는 이러한 차원에 도움이 될 수 있습니다.

- 레이턴시는 특정 종류의 트래픽 또는 애플리케이션에만 영향을 줍니까? 예: UDP, TCP, ICMP만...
- 그렇다면 이 트래픽에는 고유한 식별자가 있습니까? 예: 특정 QoS 마킹, 결정된 패킷 크기만, IP 옵션...
- 영향을 받는 사용자 또는 사이트 수 예: 하나 이상의 특정 서브넷, 하나 이상의 엔드 호스트, 하나 이상의 디바이스에 연결된 전체 사이트...
- 특정 타임스탬프가 있습니까? 예: 피크 시간, 임의의 시간 패턴 또는 완료 무작위 동안에만 발생합니까?
- 디자인 측면. 예: 특정 디바이스를 통과하는 트래픽, 많은 디바이스를 사용하지만 한 공급자에만 연결하는 트래픽, 로드 밸런싱을 수행하지만 한 경로에 영향을 주는 트래픽...

여러 가지 다른 고려 사항이 있지만 서로 다른 답변(및 이에 답하기 위해 수행할 수 있는 테스트까지)을 넘으면 효과적으로 격리하고 범위를 제한하여 트러블슈팅을 진행할 수 있습니다. 예를 들어, 서로 다른 제공자를 통과하는 모든 브랜치에서 하나의 애플리케이션(동일한 종류의 트래픽)만 영향을 받고 피크 시간에는 동일한 데이터 센터에서 종료됩니다. 이 경우 모든 지사의 모든 액세스 스위치 검사를 시작하지 않고 데이터 센터에서 더 많은 정보를 수집하는 데 주력하고 그 측면에서 더 자세히 검사합니다.

네트워크에서 사용할 수 있는 모니터링 도구 및 몇 가지 자동화는 이러한 격리에 많은 도움이 됩니다. 실제로 보유한 리소스와 고유한 상황에 따라 달라집니다.

일반적인 원인 트러블슈팅

트러블슈팅의 범위를 제한하면 특정 원인(예: 제공된 traceroute 예)을 확인하기 시작할 수 있습니다. 두 개의 다른 홉으로 격리한 다음 가능한 원인으로 범위를 좁힐 수 있습니다.

플랫폼 관련

높은 CPU

일반적인 원인 중 하나는 CPU가 높아 모든 패킷 처리에 지연이 많은 디바이스일 수 있습니다. 라우

터의 경우 라우터를 확인하는 가장 유용하고 기본적인 명령은 다음과 같습니다

라우터의 전반적인 성능:

<#root>

Router#

show platform resources

```
**State Acronym: H - Healthy, W - Warning, C - Critical
Resource                Usage                Max            Warning         Critical        State
-------------------------------------------------------------------------------------------------
RP0 (ok, active)                                                                              H

Control Processor       1.15%                100%           80%             90%             H



DRAM                    3631MB(23%)          15476MB        88%             93%             H

  bootflash             11729MB(46%)           25237MB        88%             93%             H
  harddisk              1121MB(0%)             225279MB       88%             93%             H
ESP0(ok, active)                                                                              H
 QFP                                                                                          H
  TCAM                  8cells(0%)             131072cells    65%             85%             H
  DRAM                  359563KB(1%)           20971520KB     85%             95%             H
  IRAM                  16597KB(12%)           131072KB       85%             95%             H

CPU Utilization         0.00%                100%           90%             95%             H



Crypto Utilization      0.00%                100%           90%             95%             H

  Pkt Buf Mem (0)       1152KB(0%)             164864KB       85%             95%             H
  Pkt Buf CBlk (0)      14544KB(1%)            986112KB       85%             95%             H
```

메모리 및 CPU 사용률을 한 번에 확인하는 데 유용하며, 각 CPU에 대한 임계값과 동일한 QFP(Control Plane and Data Plane)로 나뉩니다. 메모리 자체에서는 레이턴시 문제가 발생하지 않습니다. 그러나 컨트롤 플레인에 대한 DRAM 메모리가 더 이상 없으면 Cisco CEF(Express Forwarding)가 비활성화되고 CPU 사용량이 많아 레이턴시가 발생할 수 있으므로 숫자를 정상적인 상태로 유지하는 것이 중요합니다. 메모리 트러블슈팅에 대한 기본 설명서는 범위를 벗어나지만 관련 정보 섹션의 유용한 링크를 참조하십시오.

Control Processor, QFP CPU 또는 Crypto utilization에 대해 높은 CPU가 탐지되면 다음 명령을 사용할 수 있습니다.

컨트롤 플레인:

정렬된 프로세스 cpu 표시

<#root>

```
Router#
```

show processes cpu sorted

```
CPU utilization for five seconds:
```

99%/0%

```
; one minute: 13%; five minutes: 3%
 PID Runtime(ms)     Invoked      uSecs   5Sec   1Min   5Min TTY Process
  65        1621         638       2540 89.48%  1.82%  0.41%  0 crypto sw pk pro
   9         273          61       4475  1.56%  0.25%  0.05%   0 Check heaps
  51         212          64       3312  0.72%  0.21%  0.05%   0 Exec
 133         128          16       8000  0.60%  0.08%  0.01%   0 DBAL EVENTS
 473          25          12       2083  0.48%  0.04%  0.00%   0 WSMAN Process
  84        1173         353       3322  0.36%  0.07%  0.02%   0 IOSD ipc task
  87          23          12       1916  0.24%  0.02%  0.00%   0 PuntInject Keepa
  78         533         341       1563  0.12%  0.29%  0.07%   0 SAMsgThread
 225          25        1275         19  0.12%  0.00%  0.00%   0 SSS Feature Time
 386           4           4       1000  0.12%  0.00%  0.00%   0 Crypto WUI
 127         204       18810         10  0.12%  0.02%  0.00%   0 L2 LISP Punt Pro
```

컨트롤 플레인 CPU가 높은 경우(이 예제는 프로세스 때문에 99%), 프로세스를 격리해야 하며, 이에 따라 격리를 진행합니다(ARP 또는 제어 네트워크 패킷과 같은 패킷이 적용되거나 라우팅 프로토콜, 멀티캐스트, NAT, DNS, 암호화 트래픽 또는 서비스가 될 수 있음).

트래픽 흐름에 따라, 트래픽이 데이터 플레인에 집중할 수 있는 라우터로 향하지 않는 경우 추가 처리에 문제가 발생할 수 있습니다.

데이터 플레인:

show platform hardware qfp active datapath utilization [summary]

<#root>

```
Router#
```

show platform hardware qfp active datapath utilization

```
  CPP 0: Subdev 0
```

5 secs

```
        1 min        5 min       60 min
Input:  Priority (pps)            0            0            0            0
                 (bps)            0            0            0            0
    Non-Priority (pps)          231          192           68            6
                 (bps)       114616        95392        33920         3008
           Total (pps)          231          192           68            6
                 (bps)       114616        95392        33920         3008
Output: Priority (pps)            0            0            0            0
                 (bps)            0            0            0            0
```

```
    Non-Priority (pps)            3            2            2            0
                 (bps)        14896         9048         8968         2368


Total (pps)

         3323         2352          892            0


(bps)

        14896         9048         8968         2368

Processing: Load (pct)



3

            3            3            3

Crypto/IO


Crypto: Load (pct)            0

            0            0            0
        RX: Load (pct)             0            0            0            0
        TX: Load (pct)             1            1            0            0
            Idle (pct)            99           99           99           99
```

데이터 플레인이 높은 경우(처리 로드 수가 100%에 도달하는 경우) 라우터를 통과하는 트래픽의 양(초당 총 패킷 수 및 초당 비트 수)과 플랫폼의 처리량 성능을 확인해야 합니다(특정 데이터 시트에 대한 아이디어가 있을 수 있음).

이 트래픽의 예상 여부를 확인하기 위해 패킷 캡처(EPC) 또는 Netflow와 같은 모니터링 기능을 추가 분석에 사용할 수 있습니다. 몇 가지 확인 사항은 다음과 같습니다.

- 트래픽이 유효하며 이 라우터를 통과해야 합니까?
- 비정상적인 트래픽 흐름 또는 더 높은 속도를 식별합니다.
- 초당 패킷 수가 많은 경우 패킷의 크기를 확인합니다. 이 메시지가 표시되는지 또는 프래그먼트화 문제가 있는지 확인합니다.

모든 트래픽이 예상되는 경우 플랫폼 한계에 도달할 수 있습니다. 그런 다음 라우터에서 실행되는 기능을 두 번째 부분으로 찾아 show running-config를 통해 분석합니다. 대부분 인터페이스에서, identIfy 불필요한 기능을 확인하고 비활성화하거나 트래픽을 밸런싱하여 CPU 사이클을 릴리스합니다.

그러나 플랫폼 제한에 대한 표시가 없는 경우, 라우터가 패킷에 대한 지연을 추가하는지를 입증하는 또 다른 유용한 툴이 FIA 추적인지 확인할 수 있으며, 각 패킷에 대해 소요된 정확한 프로세스 시간과 대부분의 처리를 수행하는 기능을 확인할 수 있습니다. 전체 높은 CPU 문제 해결은 이 문서의 범위를 벗어나지만 관련 정보 섹션에 있는 링크를 참조하십시오.

트래픽 관련

MTU 및 단편화

MTU(Maximum Transmission Unit)는 물리적 링크가 전달할 수 있는 옥텟의 수에 따라 전송되는 최대 패킷 길이입니다.상위 레이어 프로토콜이 기본 IP에 데이터를 전송하고 IP 패킷의 결과 길이가 경로 MTU보다 클 경우 패킷은 프래그먼트로 분할됩니다. 네트워크에서 이렇게 크기가 작아지면 일부 사례에서는 처리 작업이 더 많아지고 처리 방법이 달라지므로 되도록 피해야 합니다.

NAT 또는 Zone Based Firewall과 같은 일부 기능의 경우 가상 리어셈블리는 "전체 패킷 확보"에 필요하고, 필요한 내용을 적용하고, 프래그먼트를 전달하며, 리어셈블된 사본을 폐기해야 합니다. 이 프로세스는 CPU 사이클을 추가하며 오류가 발생하기 쉽습니다.

일부 애플리케이션은 프래그먼트화에 의존하지 않습니다. MTU를 확인하는 가장 기본적인 테스트 중 하나는 no fragment 옵션을 사용하는 ping이며, 다양한 패킷 크기를 테스트합니다. ping ip-address df-bit size number입니다. ping이 실패하면 드롭이 발생하고 추가 문제가 발생할 때 경로를 통해 MTU를 수정합니다.

프래그먼트화된 패킷을 사용하는 네트워크에서 정책 기반 라우팅 및 동일 비용 다중 경로와 같은 기능을 사용하면 지연 문제가 발생할 수 있으며, 대부분 높은 데이터 속도에 더 많은 오류가 발생하여 높은 어셈블 시간, 중복 ID 및 손상된 패킷이 발생할 수 있습니다. 이러한 문제 중 일부가 파악되면 가능한 한 이 프래그먼트화를 해결해 보십시오. 프래그먼트가 있으며 잠재적인 문제가 있는지 확인하는 하나의 명령은 show ip traffic입니다.

<#root>

```
Router#
show ip traffic


IP statistics:
  Rcvd:  9875429 total, 14340254 local destination
         0 format errors, 0 checksum errors, 0 bad hop count
         0 unknown protocol, 0 not a gateway
         0 security failures, 0 bad options, 0 with options
  Opts:  0 end, 0 nop, 0 basic security, 0 loose source route
         0 timestamp, 0 extended security, 0 record route
         0 stream ID, 0 strict source route, 0 alert, 0 cipso, 0 ump
         0 other, 0 ignored


Frags:



150 reassembled
, 0
timeouts
,
0 could not reassemble
          0
fragmented
```

```
, 600

fragments

, 0

could not fragment

         0 invalid hole
  Bcast: 31173 received, 6 sent
  Mcast: 0 received, 0 sent
  Sent:  15742903 generated, 0 forwarded
  Drop:  0 encapsulation failed, 0 unresolved, 0 no adjacency
         0 no route, 0 unicast RPF, 0 forced drop, 0 unsupported-addr
         0 options denied, 0 source IP address zero
<output omitted>
```

위 출력에서 Frags 섹션의 굵은 단어는 다음을 참조하십시오.

- Reassembled(리어셈블됨): 리어셈블된 패킷 수입니다.
- 시간 초과: 패킷 조각에 대한 리어셈블 시간이 만료될 때마다.
- 리어셈블할 수 없음: 리어셈블할 수 없는 패킷 수입니다.
- Fragmented(조각화): MTU를 초과하고 조각화할 주제가 있는 패킷의 수입니다.
- Fragments(조각): 패킷이 조각화된 청크 수입니다.
- 조각화할 수 없음: MTU를 초과하지만 조각화할 수 없는 패킷 수입니다.

프래그먼트화가 사용되고 카운터가 증가하거나 카운터를 리어셈블할 수 없는 경우, 플랫폼에 의해 발생한 문제를 확인하기 위한 한 가지 방법은 QFP 삭제를 통해 수행하는 것이며, 삭제 섹션의 뒷부분에서 설명한 것과 동일한 명령을 사용합니다. show platform hardware qfp active statistics drop TcpBadfrag, IpFragErr, FragTailDrop, ReacsDrop, ReacsFragTooBig, ReacsTooManyFrags, ReacsTimeout 또는 관련 오류를 찾습니다. 각 케이스는 모든 프래그먼트, 중복, CPU 혼잡 등을 가져오지 않는 등 서로 다른 원인을 가질 수 있습니다. 다시 한 번, 추가 분석 및 잠재적 수정에 유용한 툴은 FIA 추적 및 컨피그레이션 검사가 될 수 있습니다.

TCP는 이 문제를 해결하기 위해 MSS(Maximum Segment Size) 메커니즘을 제공하지만, 잘못된 MSS 협상 또는 잘못된 경로 MTU가 검색될 경우 레이턴시를 유발할 수 있습니다.

UDP에는 이러한 프래그먼트화 메커니즘이 없으므로 PMTD 또는 임의의 애플리케이션 레이어 솔루션을 수동으로 구현하는 데 의존할 수 있으며, 프래그먼트화를 방지하기 위해 576바이트보다 짧은 패킷을 전송하도록 활성화할 수 있습니다(해당되는 경우). 이는 RFC1122에 따라 번호를 전송하는 데 필요한 유효 MTU가 더 작기 때문에 프래그먼트화를 방지할 수 있습니다.

설계 관련

이 섹션에서는 문제 해결 방법 외에도 레이턴시 문제를 가중시킬 수 있는 두 가지 주요 구성 요소에 대해 간략하게 설명하고 이 문서의 범위를 벗어나는 광범위한 논의와 분석이 필요합니다.

차선의 경로

네트워킹에서 차선의 라우팅은 데이터 패킷이 네트워크에서 사용 가능한 가장 효율적인 또는 최단 경로를 통해 전달되지 않는 상황을 의미합니다. 대신 이러한 패킷은 효율성이 떨어지는 경로를 사

용하므로 지연 시간, 정체 또는 네트워크 성능에 영향을 미칠 수 있습니다. IGP는 항상 최상의 경로를 선택합니다. 즉, 비용이 더 낮지만, 가장 저렴한 경로이거나 가장 낮은 지연 경로일 필요는 없습니다(대역폭이 더 높은 경로가 가장 좋은 경로가 될 수 있음).

라우팅 프로토콜의 문제에는 최적 이하의 라우팅이 발생할 수 있습니다. 컨피그레이션 또는 경합 조건, 동적 변경(토폴로지 변경 또는 링크 실패), 회사 정책 또는 비용을 기반으로 한 트래픽 엔지니어링, 중복 또는 장애 조치(특정 조건에서 백업 경로로 이동) 등 다른 상황에서는 최적화가 이루어지지 않습니다.

트레이스라우트 또는 모니터링 어플라이언스와 같은 툴을 사용하면 특정 플로우에 대한 이러한 상황을 파악하는 데 도움이 될 수 있으며, 이러한 경우 다른 여러 요인에 따라 애플리케이션 요구 사항을 충족하고 지연 시간을 단축하려면 라우팅 재설계 또는 트래픽 엔지니어링이 필요할 수 있습니다
.

QoS(Quality of Service)

QoS(Quality of Service)를 구성하면 다른 트래픽 유형을 희생하면서 특정 트래픽 유형을 우선적으로 처리할 수 있습니다. QoS가 없다면 디바이스 패킷 내용이나 크기에 관계없이 각 패킷에 대해 best-effort 서비스를 제공합니다. 이 디바이스 신뢰성, 지연 범위 또는 처리량을 보장하지 않고 패킷을 전송합니다.

QoS가 있는 경우 라우터 표시, 재표시 또는 패킷 분류만 수행하는지 확인하고 컨피그레이션을 확인하고 policy-map [name_of_policy_map을 표시하는 것이 매우 중요합니다 | 세션 | interface interface_id]는 높은 비율, 삭제 또는 잘못 분류된 패킷의 영향을 받는 클래스를 이해하는 데 도움이 됩니다.

QoS 구현은 심각한 분석이 필요한 중대한 작업이며 이 문서의 범위에 속하지 않지만, 시간에 민감한 애플리케이션의 우선 순위를 정하고 많은 지연 시간과 애플리케이션 문제를 해결하거나 방지하기 위해 이를 고려하는 것이 좋습니다.

기타 성능 문제

다른 조건은 속도 저하, 세션 재연결 또는 일반적인 불량 성능을 확인해야하는 경우를 추가할 수 있습니다. 그 중 일부는 다음과 같습니다.

삭제

디바이스에서 처리와 직접 관련된 문제는 패킷 삭제입니다. 인터페이스 관점에서 입력 및 출력 측면을 확인해야 합니다.


```
Router#sh interfaces GigabitEthernet0/0/1
GigabitEthernet0/0/1 is up, line protocol is up
  Hardware is vNIC, address is 0ce0.995d.0000 (bia 0ce0.995d.0000)
  Internet address is 10.10.1.2/24
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
```

```
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full Duplex, 1000Mbps, link type is auto, media type is Virtual
  output flow-control is unsupported, input flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:19, output 00:08:33, output hang never
  Last clearing of "show interface" counters never
```

Input queue: 0/375/6788/0 (size/max/drops/flushes); Total output drops: 18263

```
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 114000 bits/sec, 230 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     193099 packets input, 11978115 bytes, 0 no buffer
     Received 0 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
```

1572 input errors

,

12 CRC

, 0 frame,

1560 overrun

```
, 0 ignored
     0 watchdog, 0 multicast, 0 pause input
     142 packets output, 11822 bytes, 0 underruns
     Output 0 broadcasts (0 IP multicasts)
     0 output errors, 0 collisions, 0 interface resets
     23 unknown protocol drops
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 pause output
     0 output buffer failures, 0 output buffers swapped out
Router#
```

입력측에 제공되는 항목은 다음과 같습니다.

- 입력 대기열 삭제: 각 인터페이스는 라우팅 프로세서(RP)에 의한 처리를 대기하도록 수신 패킷이 배치되는 입력 대기열(수정할 수 있는 소프트웨어 버퍼)을 소유합니다. 입력 대기열에 배치된 수신 패킷의 속도가 RP가 패킷을 처리할 수 있는 속도를 초과하는 경우 드랍이 증가할 수 있습니다. 그러나 제어 패킷과 "For us" 트래픽만 배치되므로, 산발적인 드롭이 있더라도 트래픽 통과 시 레이턴시가 발견되면 이러한 문제가 발생하지 않아야 합니다.

- 오버런: 입력 속도가 데이터를 처리하는 수신기의 능력을 초과하기 때문에 수신기 하드웨어가 수신된 패킷을 하드웨어 버퍼에 전달할 수 없을 때 발생합니다. 이 숫자는 라우터의 속도 및 성능에 문제가 있음을 나타낼 수 있으며, 이 인터페이스에 대해서만 트래픽을 캡처하고 트래픽 스파이크를 검색합니다. 일반적인 해결 방법은 플로우 제어를 활성화하는 것이지만, 이를 통해 패킷을 지연시킬 수 있습니다. 이는 병목 현상과 초과 유입의 증거이기도 합니다.
- CRC: 물리적 문제로 인해 발생하며 케이블, 포트 및 SFP가 올바르게 연결되었고 제대로 작동하는지 확인합니다.

출력에는 다음이 포함됩니다.

- 출력 대기열 삭제: 각 인터페이스는 출력 대기열을 소유합니다. 여기서 는 인터페이스에서 전송할 발신 패킷이 배치됩니다. RP에 의해 출력 대기열에 배치된 발신 패킷의 비율이 인터페이스가 패킷을 전송할 수 있는 비율을 초과하는 경우도 있습니다. 이로 인해 QoS가 없는 경우 성능 문제 및 레이턴시 문제가 발생할 수 있습니다. 그렇지 않으면 특정 정책이 적용되어 이 수가 증가할 수 있으며 의도되거나 중요한 트래픽을 보호하고 보장하기 위해 QoS 컨피그레이션을 확인하거나 구현하도록 조언할 수 있습니다.

마지막으로, QFP의 삭제는 레이턴시를 야기할 수 있는 높은 처리와 직접적으로 관련이 있습니다. show platform hardware qfp active statistics drop을 통해 확인하십시오.

<#root>

```
Router#

show platform hardware qfp active statistics drop

Last clearing of QFP drops statistics : never

-------------------------------------------------------------------------------
Global Drop Stats                                Packets              Octets
-------------------------------------------------------------------------------
Disabled                                               2                 646
Ipv4NoAdj                                         108171             6706602
Ipv6NoRoute                                           10                 560
```

FIA 추적은 코드에 따라 다르며, 레이턴시의 영향을 받는 트래픽이 이 지점에서 삭제되는 경우 확증하거나 폐기하는 데 도움이 됩니다.

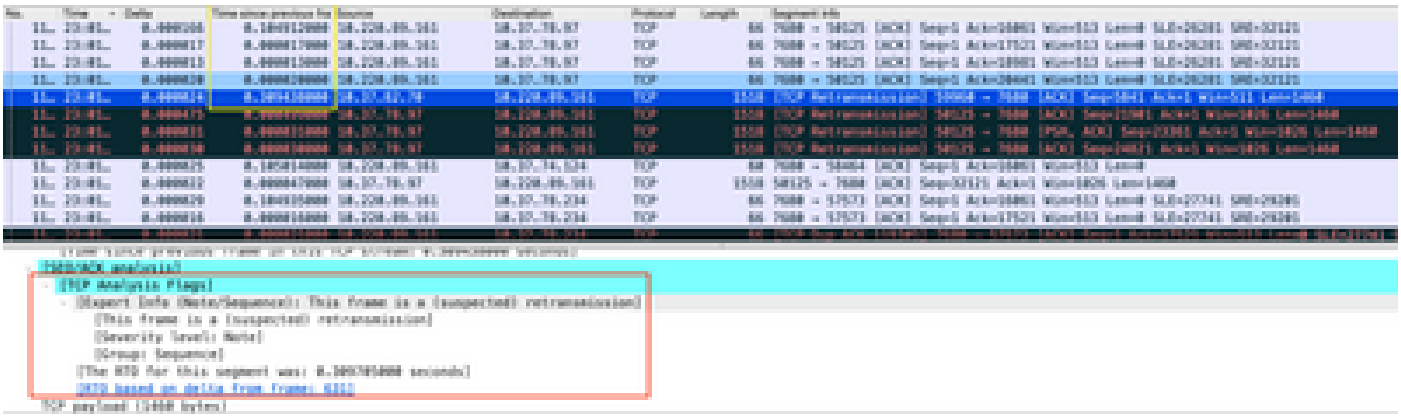
TCP 재전송

TCP 재전송은 증상이거나 패킷 손실과 같은 언더레이 문제로 인해 발생할 수 있습니다. 이 문제는 응용 프로그램에서 느려지고 성능이 저하되는 것을 유발할 수 있습니다.

TCP(Transmission Control Protocol)는 원격 데이터 수신기로부터의 피드백이 없는 경우 재전송 타이머를 사용하여 데이터 전달을 보장합니다. 이 타이머의 지속 시간을 RTO(재전송 시간 제한)라고 합니다. 재전송 타이머가 만료되면, 발신자는 TCP 수신자가 확인하지 못한 가장 빠른 세그먼트를 재전송하며 RTO가 증가된다.

일부 재전송은 완전히 제거할 수 없으며, 최소한이면 문제를 반영할 수 없습니다. 그러나 유추할 수 있듯이 더 많은 재전송이 확인되고 TCP 세션의 레이턴시가 늘어나므로 이를 해결해야 합니다.

Wireshark에서 분석한 패킷 캡처는 다음 예제와 같이 이 문제를 확증할 수 있습니다.



TCP 대화 캡처

재전송이 있는 경우, 라우터 인그레스 및 이그레스 방향에서 동일한 캡처 방법을 사용하여 전송 및 수신된 모든 패킷을 확인합니다. 물론, 모든 홉에서 이 작업을 수행하는 것은 엄청난 작업이므로 TCP에 대해 캡처에 대한 자세한 분석이 필요합니다. TTL, 동일한 TCP 스트림의 이전 프레임에서 보낸 시간을 보면 어떤 방향(서버 또는 클라이언트)에서 이러한 지연 또는 트러블슈팅을 지시할 수 있는 응답이 부족한지 알 수 있습니다.

초과 유입 및 병목 현상

필요한 리소스(대역폭)가 실제 사용 가능한 리소스보다 클 경우 초과 서브스크립션이 발생합니다. 라우터에 이 문제가 있는지 확인하는 명령은 이전 섹션에서 이미 다뤘습니다.

이러한 상황으로 인해 대역폭 또는 하드웨어 용량 부족으로 인해 트래픽 흐름이 느려질 경우 병목 현상이 발생할 수 있습니다. 단기간에 이런 일이 일어나는지, 아니면 솔루션을 적용하기에는 장기적인 상황인지를 파악하는 것이 중요하다.

이를 해결하기 위한 구체적인 조언은 없지만, 일부 옵션은 현재 요구 사항 및 향후 성장 분석을 기반으로 다른 플랫폼으로의 트래픽 균형 조정, 네트워크 분할 또는 보다 강력한 디바이스로의 업그레이드에 있습니다.

관련 정보

- IP SLA ICMP 에코 작업
- 메모리 문제 해결
- Cisco IOS-XE Datapath 패킷 추적 기능으로 문제 해결
- ASR 1000 Series 서비스 라우터의 패킷 삭제 문제를 해결합니다.
- Qos 관련 정보
- 라우터의 QoS 컨피그레이션
- Cisco 기술 지원 및 다운로드

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.