在Cisco IOS®；上配置TCP最佳化功能XE SD-WAN cEdge路由器

目錄

簡介

本檔案介紹Cisco IOS® XE SD-WAN路由器上的傳輸控制通訊協定(TCP)最佳化功能，該功能於2019年8月在16.12版本中引入。涵蓋的主題包括先決條件、問題描述、解決方案、Viptela OS(vEdge)和XE SD-WAN(cEdge)之間的TCP最佳化演算法差異、配置、驗證和相關文檔清單。

必要條件

需求

本文件沒有特定需求。

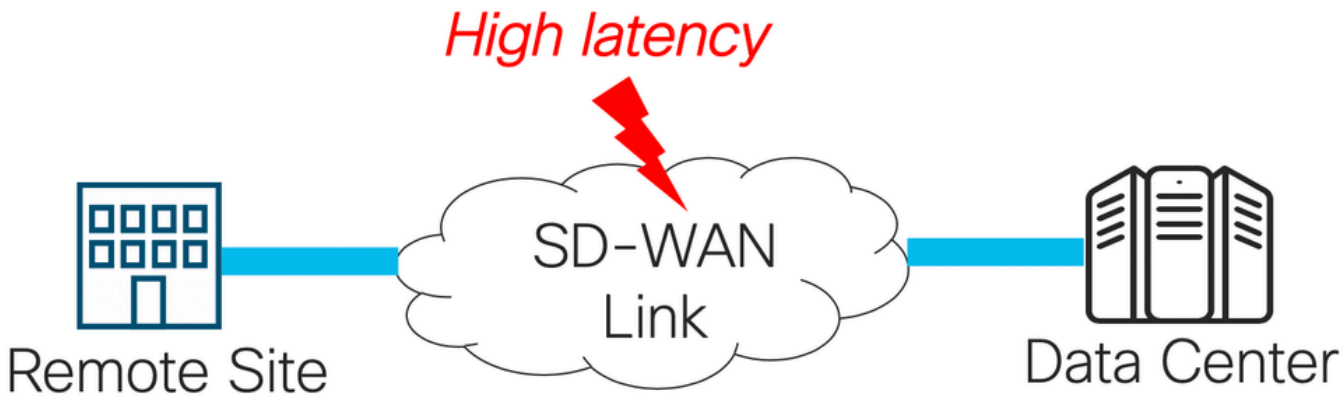
採用元件

本檔案中的資訊是根據Cisco IOS® XE SD-WAN。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

問題

兩個SD-WAN端之間的WAN鏈路上的高延遲會導致應用程式效能下降。您有重要的TCP流量，必須對其進行最佳化。



解決方案

使用TCP最佳化功能時，可以改善兩個SD-WAN站點之間關鍵TCP流的平均TCP吞吐量。

瞭解cEdge瓶頸頻寬和往返傳輸(BBR)與vEdge(CUBIC)的TCP最佳化之間的異同

快速BBR傳播時間演算法用於XE SD-WAN實現（在cEdge上）。

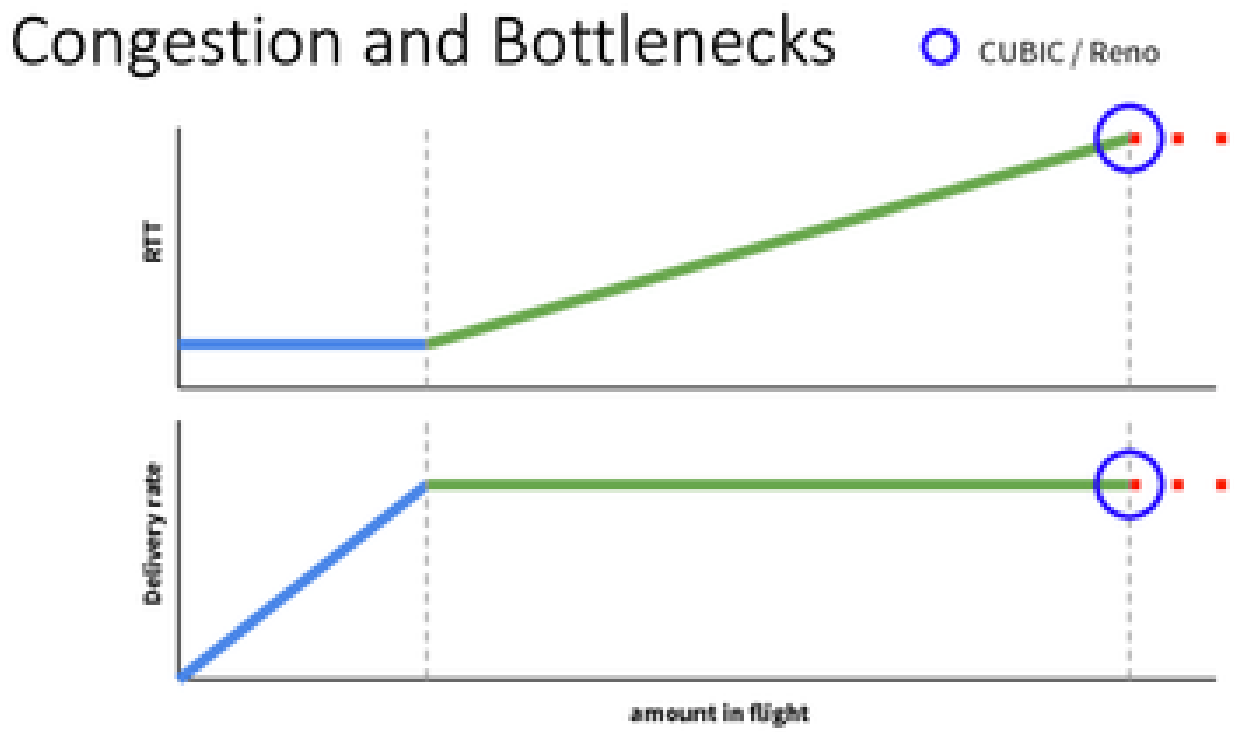
Viptela OS(vEdge)有一個不同的、較舊的演算法，稱為CUBIC。

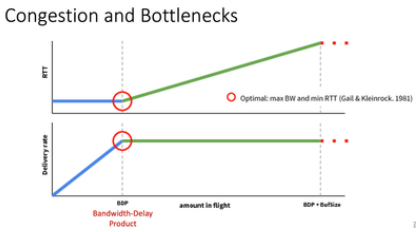
CUBIC主要考慮丟包問題，並廣泛應用於不同的客戶端作業系統。Windows、Linux、MacOS、Android已經內建了CUBIC。在某些情況下，如果舊客戶端運行的是不帶CUBIC的TCP堆疊，則在vEdge上啟用TCP最佳化會帶來改進。vEdge TCP CUBIC最佳化受益的其中一個例子是使用舊客戶端主機和WAN鏈路出現嚴重延遲/丟棄的潛水艇。請注意，只有vEdge 1000和vEdge 2000支援TCP CUBIC。

BBR主要關注往返時間和延遲。在丟包時不會。如果您通過公共Internet從美國西部向東海岸甚至是歐洲傳送資料包，在大多數情況下您不會看到任何資料包丟失。公共Internet有時在丟包方面表現過於出色。但是，您看到的是延遲/延遲。這個問題由BBR解決，BBR由Google於2016年開發。

簡言之，BBR對網路建模，並檢視每個確認(ACK)並更新最大頻寬(BW)和最小往返時間(RTT)。 然後基於模型控制傳送：探查最大BW和最小RTT，接近估計BW並保持傳入接近頻寬延遲產品(BDP)。 主要目標是通過較小的瓶頸隊列確保高吞吐量。

Mark Claypool中的此幻燈片顯示了CUBIC操作的區域：





BBR的運行環境更佳，Mark Claypool在此幻燈片中也顯示了這一點：

如果您想瞭解有關BBR 演算法的更多資訊，可以在此處的bbr-dev郵件清單首頁的頂部找到多個連結有關BBR的發佈檔案。

總而言之：

平台和演算法	鍵輸入引數	使用案例
cEdge(XE SD-WAN):BBR	RTT/延遲	兩個SD-WAN站點之間的關鍵TCP流量
vEdge（Viptela作業系統）：CUBICP	封包遺失	沒有任何TCP最佳化的舊客戶端

支援的XE SD-WAN平台

在XE SD-WAN軟體版本16.12.1d中，這些cEdge平台支援TCP最佳化BBR:

- ISR4331
- ISR4351
- CSR1000v，帶8個vCPU且最少8 GB RAM

注意事項

- 當前不支援DRAM小於8 GB RAM的所有平台。
- 目前不支援具有4個或更少資料核心的所有平台。
- TCP最佳化不支援MTU 2000。
- 當前 — 不支援IPv6流量。
- 不支援對流向第三方BBR伺服器的DIA流量進行最佳化。您需要在兩端都安裝一台cEdge SD-WAN路由器。
- 在當前的資料中心方案中，每個控制節點(CN)僅支援一個服務節點(SN)。
- 16.12版本不支援在同一裝置上同時使用具有安全性（UTD容器）和TCP最佳化的使用案例。從17.2開始支援組合使用情形。操作順序為TCP Opt先，然後是安全。此封包將傳送到TCP最佳化容器中，然後將服務鏈結到UTD安全容器中（沒有第二次傳送）。TCP最佳化是針對整個流進行的，而不是針對前幾個位元組。如果UTD丟棄，則整個連線被丟棄。

附註：ASR1k當前不支援TCP最佳化。但是，有一個針對ASR1k的解決方案，其中ASR1k通過AppNav隧道（GRE封裝）將TCP流量傳送到外部CSR1kv以進行最佳化。目前（2020年2月）僅支援一個CSR1k作為單個外部服務節點，但未經過充分測試。稍後將在「配置」部分對此進行說明。

下表總結了每個版本的警告，並強調了支援的硬體平台：

案例	使用案例	16.12.1	17.2.1	17.3.1	17.4.1	意見
分支機構到Internet	DIA	否	是	是	是	在16.12.1中，網際網路介面上未啟用AppQoE FIA
	SAAS	否	是	是	是	在16.12.1中，網際網路介面上未啟用AppQoE FIA
分支機構到DC	單邊緣路由器	否	否	否	是	需要支援多個SN
	多邊緣路由器	否	否	否	是	需要流對稱或Appnav流同步。16.12.1未經測試
	多個SN	否	否	否	是	vManage增強功能，可接受多個SN IP
分支機構到分支機構	全網狀網路（輻條到輻條）	是	是	是	是	

	中心輻射型 （輻條 — 中心輻條）	否	是	是	是	
BBR支援	含BBR的TCP光纖	部分	部分	完整版	完整版	
平台	支援的平台	只有4300和CSR	除ISR1100外的所有產品	All	All	

設定

SN和CN的概念用於TCP最佳化：

- SN是一個守護進程，負責實際最佳化TCP資料流。
- CN稱為AppNav Controller，負責流量選擇以及往返於SN的傳輸。

SN和CN可以在同一路由器上運行，也可以作為不同的節點分開運行。

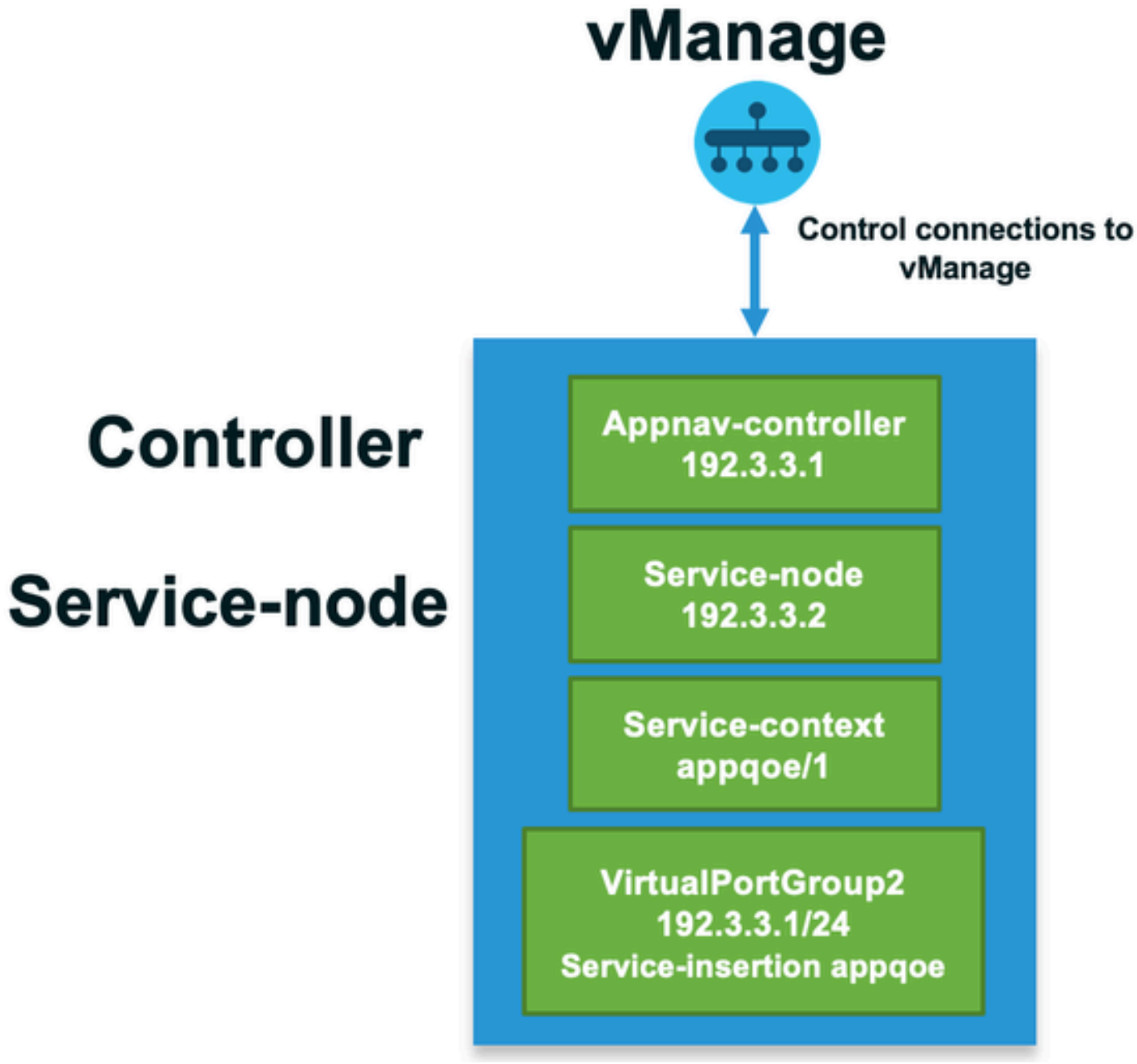
有兩種主要用例：

1. SN和CN運行在同一ISR4k路由器上的分支機構使用案例。
2. 資料中心使用案例，其中CN在ASR1k上運行，SN在單獨的CSR1000v虛擬路由器上運行。

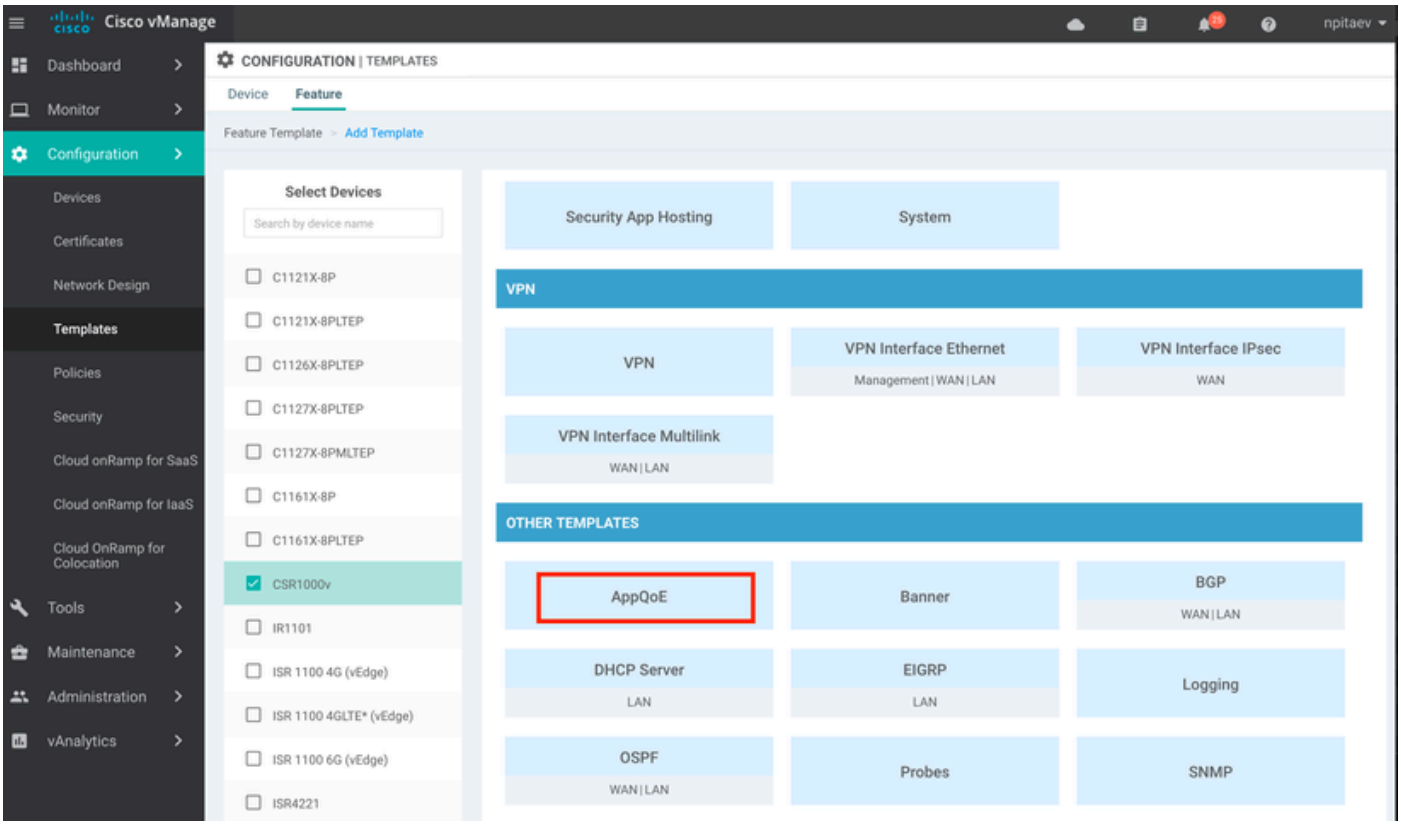
這兩種使用情形都在本節中描述。

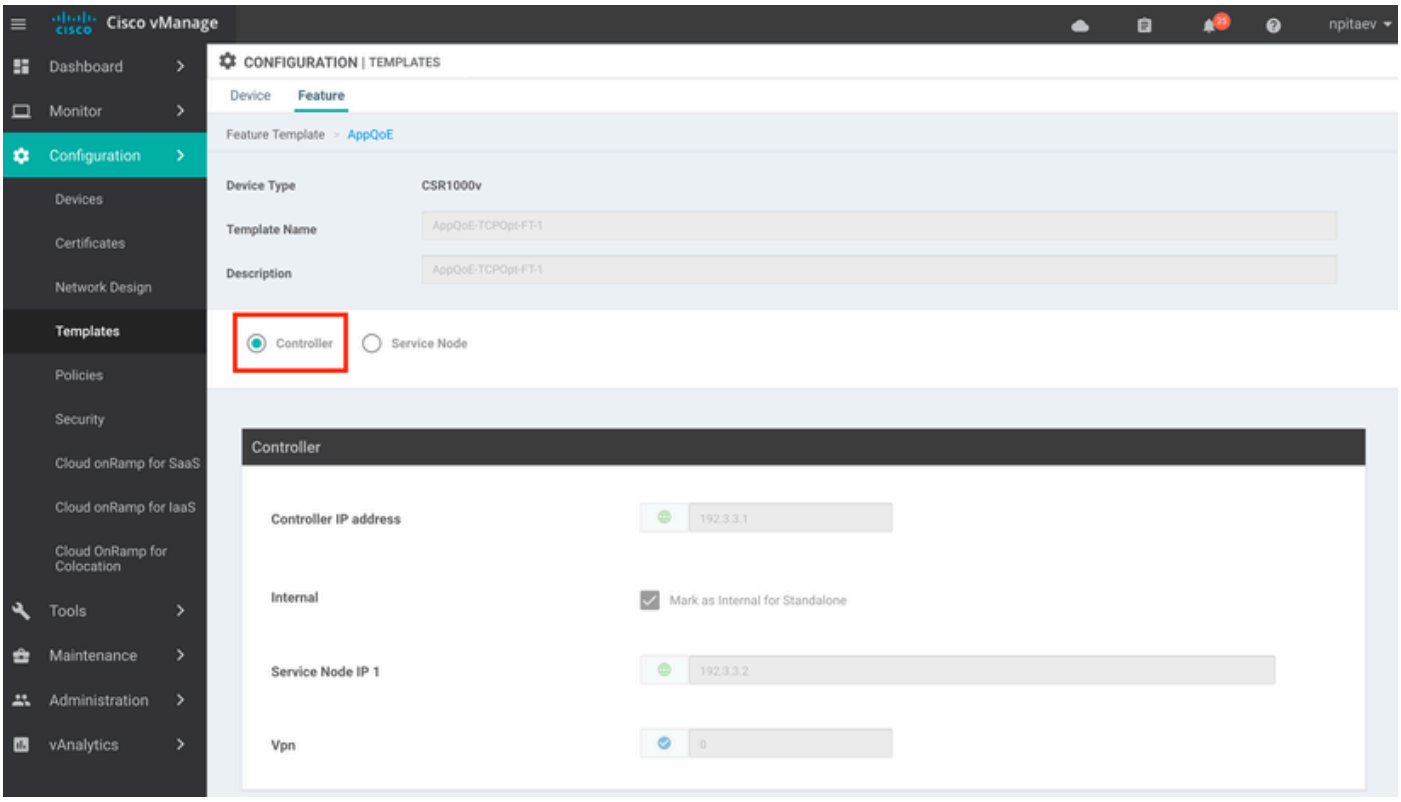
使用案例1.在分支上配置TCP最佳化（全部在一個cEdge中）

此圖顯示了分支機構單個獨立選項的整體內部架構：

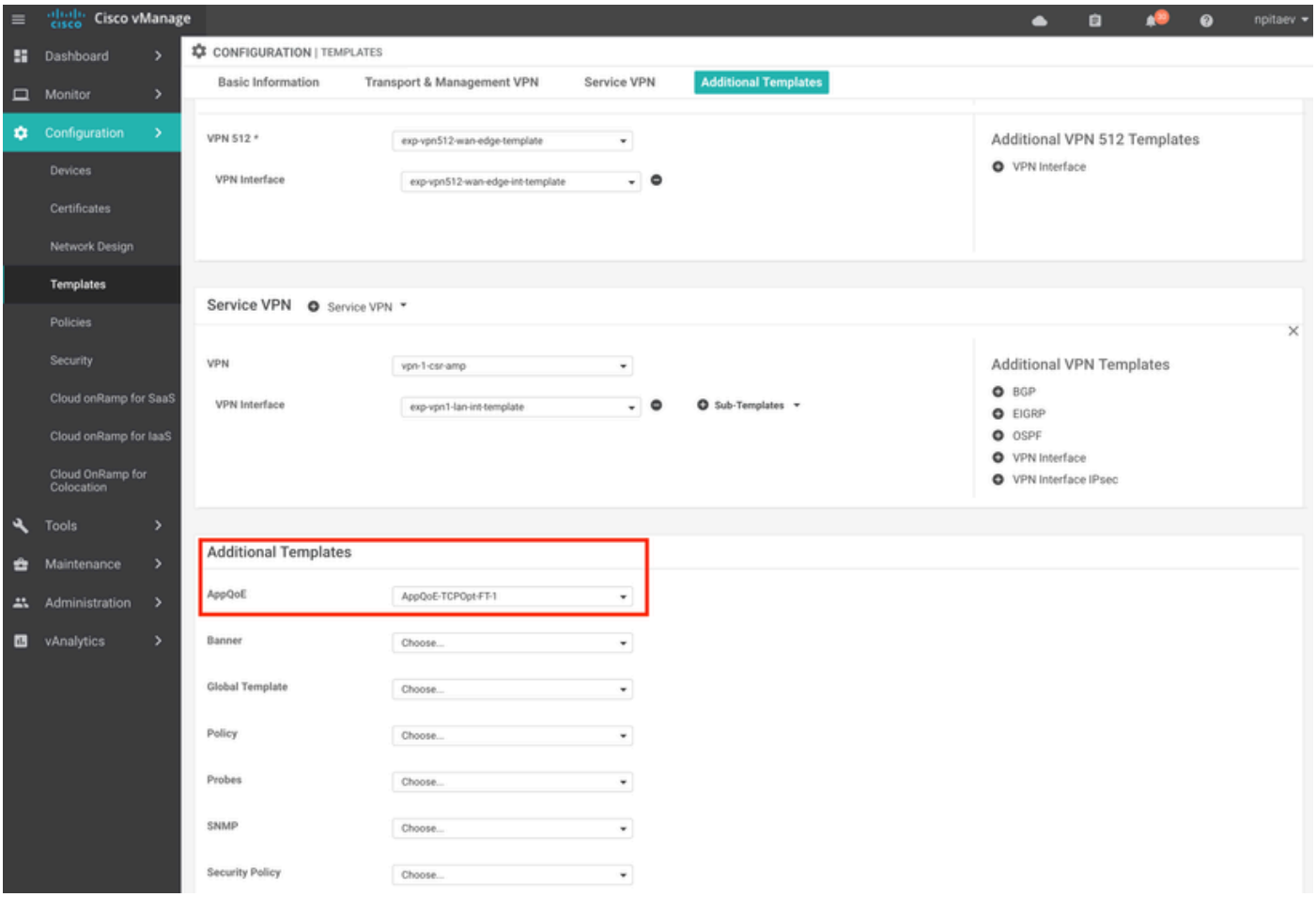


步驟1。若要設定TCP最佳化，需要在vManage中為TCP最佳化建立一個功能模板。導覽至Configuration > Templates > Feature Templates > Other Templates > AppQoE，如下圖所示。





步驟2.將AppQoE功能模板新增到Additional Templates下的相應裝置模板中：

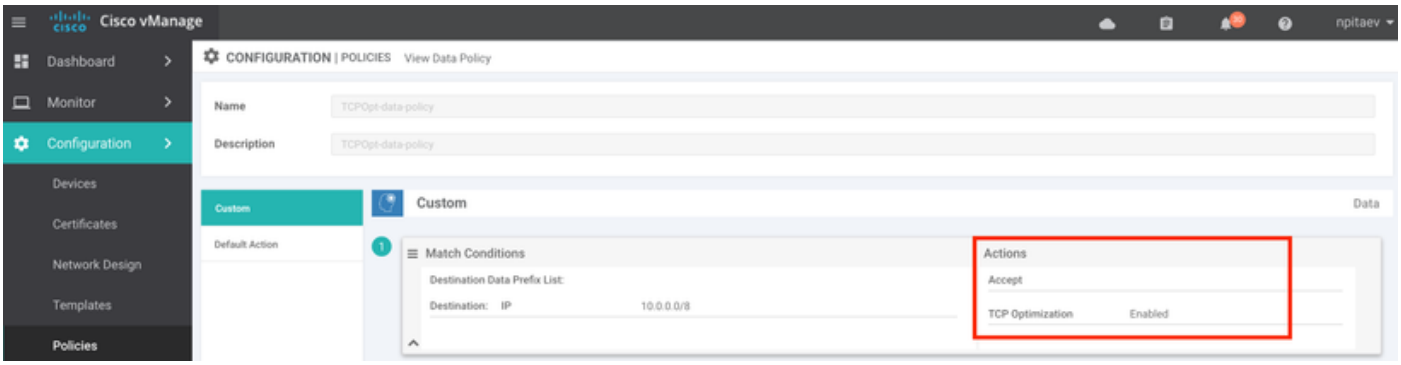


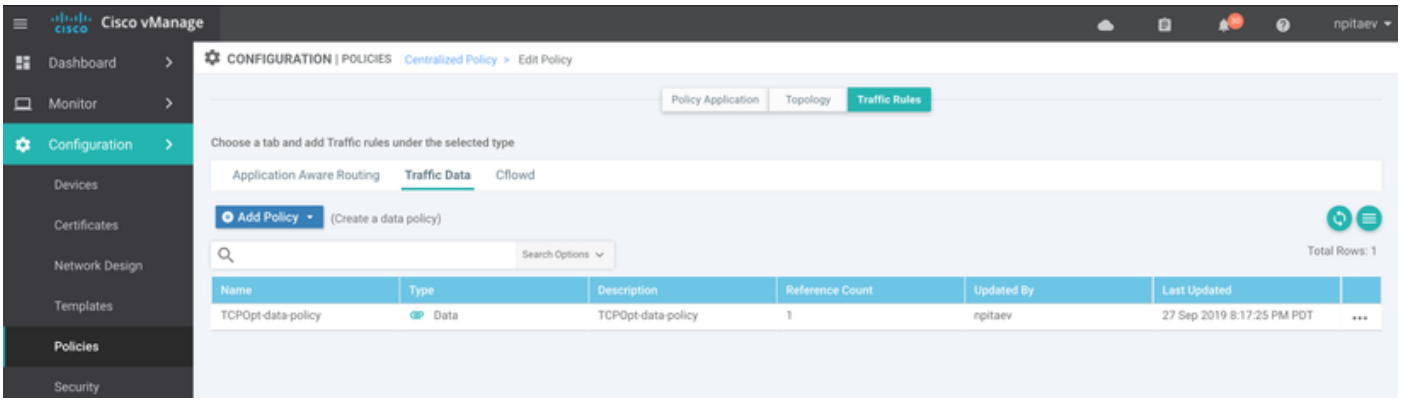
以下是模板配置的CLI預覽：

```
service-insertion service-node-group appqoe SNG-APPQOE
 service-node 192.3.3.2
!
service-insertion appnav-controller-group appqoe ACG-APPQOE
 appnav-controller 192.3.3.1
!
service-insertion service-context appqoe/1
 appnav-controller-group ACG-APPQOE
 service-node-group SNG-APPQOE
 vrf global
 enable
!
!
interface VirtualPortGroup2
 ip address 192.3.3.1 255.255.255.0
 no mop enabled
 no mop sysid
 service-insertion appqoe
!
```

步驟3.使用所關心的TCP流量的定義建立集中資料策略以進行最佳化。

舉個例子；此資料策略匹配IP字首10.0.0.0/8（包括源地址和目標地址），並為它啟用TCP最佳化：





以下是vSmart策略的CLI預覽：

<#root>

```
policy
 data-policy _vpn-list-vpn1_TCPOpt_1758410684
  vpn-list vpn-list-vpn1
    sequence 1
     match
      destination-ip 10.0.0.0/8
     !
     action accept


tcp-optimization

     !
    !
  default-action accept
 !
 lists
  site-list TCPOpt-sites
   site-id 211
   site-id 212
  !
  vpn-list vpn-list-vpn1
   vpn 1
  !
 !
!
apply-policy
 site-list TCPOpt-sites
  data-policy _vpn-list-vpn1_TCPOpt_1758410684 all
 !
!
```

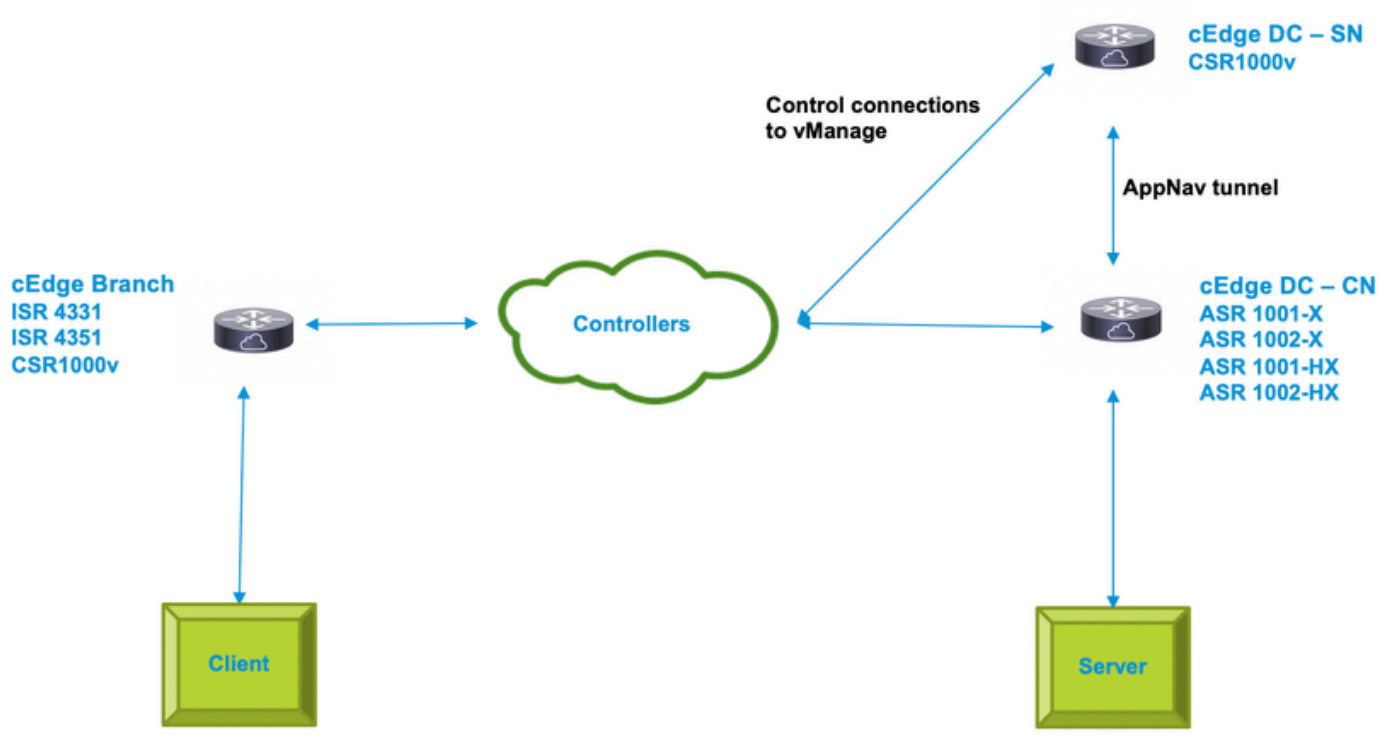
使用案例2.使用外部SN配置資料中心中的TCP最佳化

SN和CN的物理分離是兩種分支使用情形的主要區別。在多功能一體分支機構使用案例中，連線是在同一路由器內使用虛擬埠組介面完成的。在資料中心使用案例中，充當CN的ASR1k與作為SN運行的外部CSR1k之間存在AppNav GRE封裝隧道。CN和外部SN之間不需要專用的鏈路或交叉連線，簡單IP可達性就足夠了。

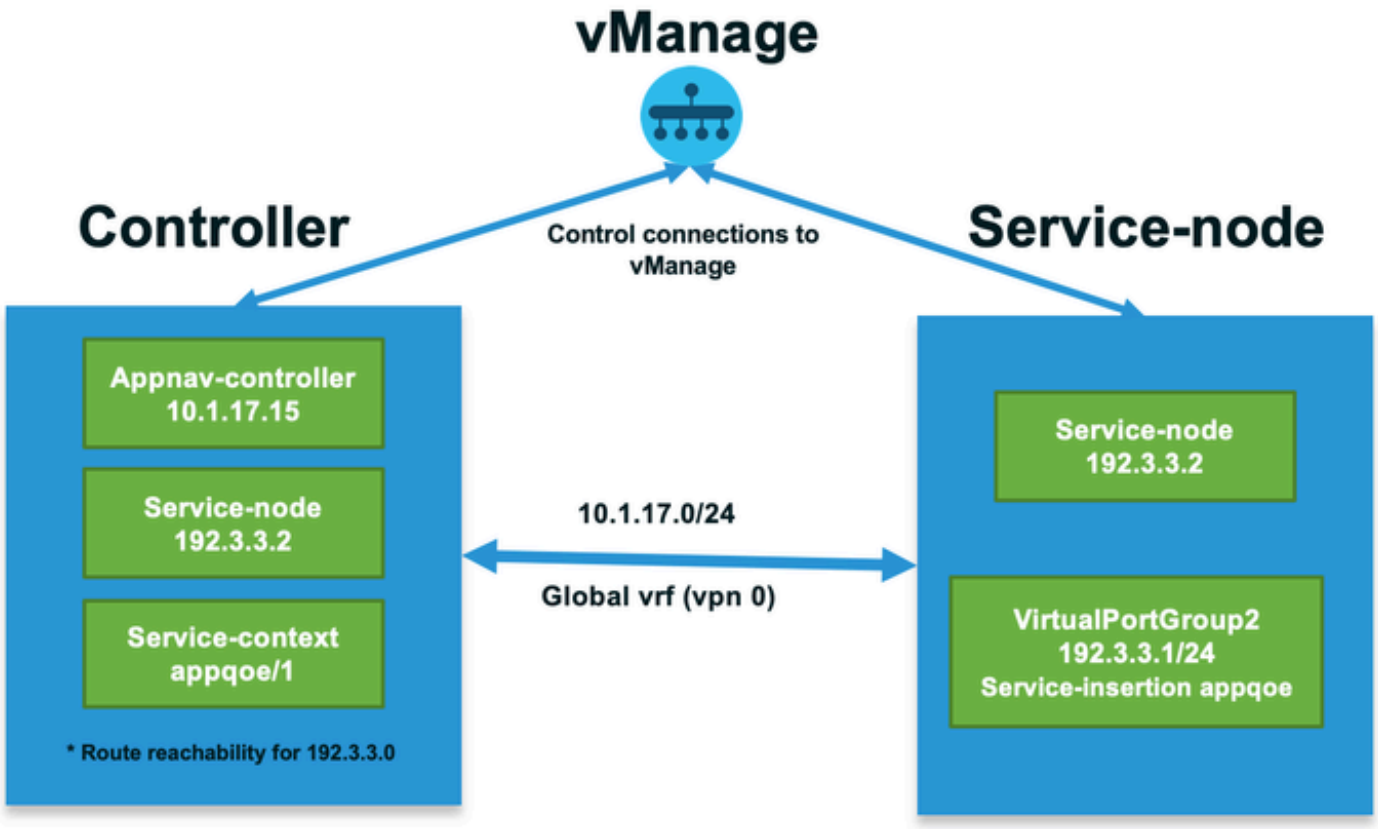
每個SN有一個AppNav(GRE)隧道。為了將來使用（支援多個SN），建議對CN和SN之間的網路使用/28子網。

建議在CSR1k上使用2個NIC作為SN。如果SN必須由vManage配置/管理，則需要SD-WAN控制器的第2個NIC。如果要手動配置/管理SN，則第二個NIC是可選的。

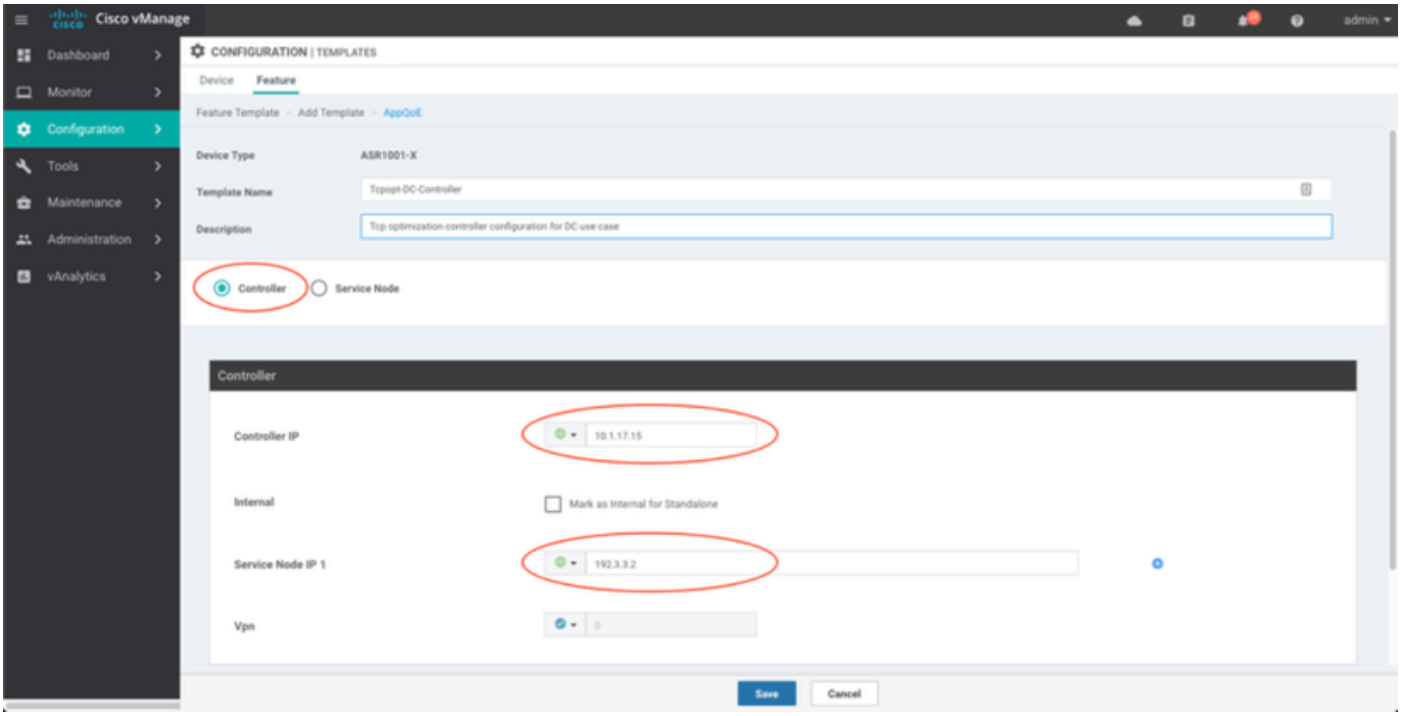
此圖顯示資料中心ASR1k作為CN運行，CSR1kv作為服務節點SN運行：



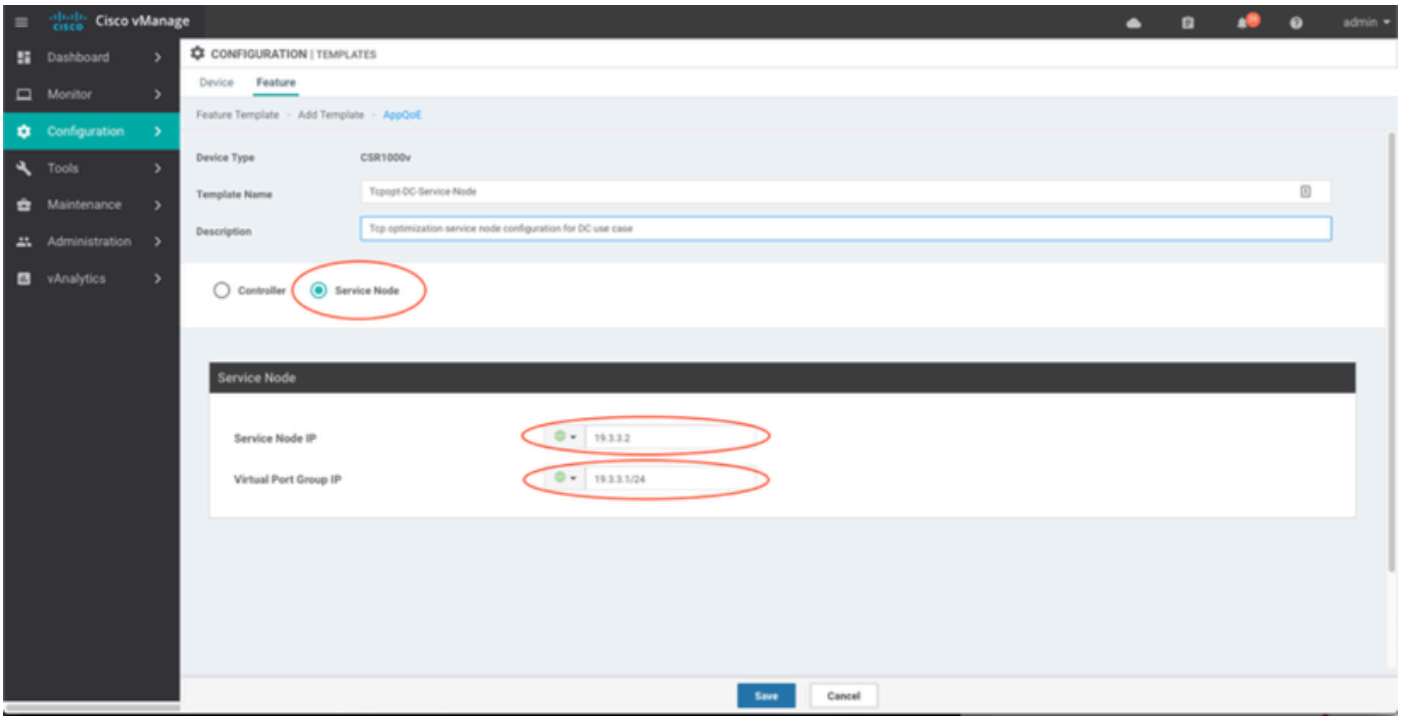
ASR1k和外部CSR1k的資料中心使用案例的拓撲如下所示：



此AppQoE功能模板顯示ASR1k配置為控制器：



配置為外部服務節點的CSR1k如下所示：



故障轉移案例

在資料中心使用案例中，如果外部CSR1k出現故障，則CSR1k充當SN，則進行故障轉移：

- 由於SN上的TCP會話被終止，已存在的TCP會話將丟失。
- 新的TCP會話將傳送到最終目的地，但TCP流量未最佳化（繞過）。
- 如果SN發生故障，則不會對相關流量進行黑洞。

故障切換檢測基於AppNav心跳，即每秒1次。在3或4個錯誤後，隧道被宣告為關閉。

分支機構使用情形中的故障切換類似 — 在SN出現故障的情況下，會將未最佳化的流量直接傳送到目的地。

驗證

使用本節內容，確認您的組態是否正常運作。

使用此CLI命令驗證CLI上的TCP最佳化，並檢視最佳化流程摘要：

```
<#root>

BR11-CSR1k#

show plat hardware qfp active feature sdwan datapath appqoe summary

TCPOPT summary
----------------

  optimized flows     : 2

  expired flows       : 6033
  matched flows       : 0
  divert pkts         : 0
```

```
  bypass pkts        : 0
  drop pkts          : 0
  inject pkts        : 20959382
  error pkts         : 88

BR11-CSR1k#
```

此輸出提供了有關最佳化流的詳細資訊：

<#root>

```
BR11-CSR1k#
```

show platform hardware qfp active flow fos-to-print all

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
     GLOBAL CFT  ~  Max Flows:2000000   Buckets Num:4000000
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Filtering parameters:
    IP1 : ANY
    Port1 : ANY
    IP2 : ANY
    Port2 : ANY
    Vrf id : ANY
    Application: ANY
    TC id: ANY
    DST    Interface id: ANY
    L3 protocol : IPV4/IPV6
    L4 protocol : TCP/UDP/ICMP/ICMPV6
    Flow type : ANY
Output parameters:
    Print CFT internal data ? No
    Only print summary ? No
    Asymmetric : ANY
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
keyID: SrcIP SrcPort DstIP DstPort L3-Protocol L4-Protocol vrfID
====================================================================
   key #0: 192.168.25.254 26113 192.168.25.11 22 IPv4 TCP 3
   key #1: 192.168.25.11 22 192.168.25.254 26113 IPv4 TCP 3
====================================================================
   key #0: 192.168.25.254 26173 192.168.25.11 22 IPv4 TCP 3
   key #1: 192.168.25.11 22 192.168.25.254 26173 IPv4 TCP 3
====================================================================
   key #0: 10.212.1.10 52255 10.211.1.10 8089 IPv4 TCP 2
   key #1: 10.211.1.10 8089 10.212.1.10 52255 IPv4 TCP 2

Data for FO with id: 2
------------------------
```

appqoe

```
: flow action DIVERT, svc_idx 0, divert pkt_cnt 1, bypass pkt_cnt 0, drop pkt_cnt 0, inject pkt_cnt 1,
====================================================================
   key #0: 10.212.1.10 52254 10.211.1.10 8089 IPv4 TCP 2
   key #1: 10.211.1.10 8089 10.212.1.10 52254 IPv4 TCP 2

Data for FO with id: 2
------------------------
```

appqoe

```
: flow action DIVERT, svc_idx 0, divert pkt_cnt 158, bypass pkt_cnt 0, drop pkt_cnt 0, inject pkt_cnt 2
=====================================================================
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
           Number of flows that passed filter: 4
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                        FLOWS DUMP DONE.
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

BR11-CSR1k#
```

疑難排解

以下CLI將有助於確定特定TCP流的問題。

所有示例均取自ISR4431上運行的IOS XE SD-WAN 17.2.1映像。

1. 查詢VRF ID（與VRF名稱不同）。 在此案例中，我們在以下輸出中看到，服務VPN(VRF)1具有VRF Id 2:

<#root>

```
AppQoE_R2#
```

show vrf detail

```
VRF 1 (
```

VRF Id = 2

```
); default RD 1:1; default VPNID <not set>
  New CLI format, supports multiple address-families
  Flags: 0x180C
  Interfaces:
    Gi0/0/3
…
```

2. 查詢服務vpn中客戶端的相應流ID — 我們在此處看到使用TCP埠445上的SMB共用的兩個Windows客戶端之間的4個不同流：

<#root>

```
AppQoE_R2#
```

show sdwan appqoe flow vpn-id 2 client-ip 192.168.200.50

```
Optimized Flows
---------------
T:TCP, S:SSL, U:UTD
Flow ID      VPN  Source IP:Port        Destination IP:Port   Service
15731593842  2    192.168.200.50:49741  192.168.100.50:445    T
17364128987  2    192.168.200.50:49742  192.168.100.50:445    T
```

```
25184244867  2    192.168.200.50:49743  192.168.100.50:445    T
28305760200  2    192.168.200.50:49744  192.168.100.50:445    T
AppQoE_R2#
```

3. 請參閱相應流的TCP Opt統計資訊：

<#root>

AppQoE_R2#

show sdwan appqoe flow flow-id 15731593842

```
VPN: 2  APP: 0 [Client 192.168.200.50:49741 - Server 192.168.100.50:445]

TCP stats
---------
Client Bytes Received   : 14114
Client Bytes Sent       : 23342
Server Bytes Received   : 23342
Server Bytes Sent       : 14114
TCP Client Rx Pause     : 0x0
TCP Server Rx Pause     : 0x0
TCP Client Tx Pause     : 0x0
TCP Server Tx Pause     : 0x0
Client Flow Pause State : 0x0
Server Flow Pause State : 0x0
TCP Flow Bytes Consumed : 0
TCP Client Close Done   : 0x0
TCP Server Close Done   : 0x0
TCP Client FIN Rcvd     : 0x0
TCP Server FIN Rcvd     : 0x0
TCP Client RST Rcvd     : 0x0
TCP Server RST Rcvd     : 0x0
TCP FIN/RST Sent        : 0x0
Flow Cleanup State      : 0x0
TCP Flow Events
  1. time:2196.550604   ::   Event:TCPPROXY_EVT_FLOW_CREATED
  2. time:2196.550655   ::   Event:TCPPROXY_EVT_SYNCACHE_ADDED
  3. time:2196.552366   ::   Event:TCPPROXY_EVT_ACCEPT_DONE
  4. time:2196.552665   ::   Event:TCPPROXY_EVT_CONNECT_START
  5. time:2196.554325   ::   Event:TCPPROXY_EVT_CONNECT_DONE
  6. time:2196.554370   ::   Event:TCPPROXY_EVT_DATA_ENABLED_SUCCESS

AppQoE_R2#
```

4. 另請參閱通用TCP選擇統計資訊，其中還顯示了4個最佳化流：

<#root>

AppQoE_R2#

show tcpproxy statistics

```
==========================================================
```

TCP Proxy Statistics

```
============================================================
Total Connections                    : 6
Max Concurrent Connections           : 4
Flow Entries Created                 : 6
Flow Entries Deleted                 : 2

Current Flow Entries                 : 4


Current Connections                  : 4


Connections In Progress              : 0
Failed Connections                   : 0
SYNCACHE Added                       : 6
SYNCACHE Not Added:NAT entry null    : 0
SYNCACHE Not Added:Mrkd for Cleanup  : 0
SYN purge enqueued                   : 0
SYN purge enqueue failed             : 0
Other cleanup enqueued               : 0
Other cleanup enqueue failed         : 0
Stack Cleanup enqueued               : 0
Stack Cleanup enqueue failed         : 0
Proxy Cleanup enqueued               : 2
Proxy Cleanup enqueue failed         : 0
Cleanup Req watcher called           : 135003
Total Flow Entries pending cleanup   : 0
Total Cleanup done                   : 2
Num stack cb with null ctx           : 0
Vpath Cleanup from nmrx-thread       : 0
Vpath Cleanup from ev-thread         : 2
Failed Conn already accepted conn    : 0
SSL Init Failure                     : 0

Max Queue Length Work                : 1
Current Queue Length Work            : 0
Max Queue Length ISM                 : 0
Current Queue Length ISM             : 0
Max Queue Length SC                  : 0
Current Queue Length SC              : 0
Total Tx Enq Ign due to Conn Close   : 0
Current Rx epoll                     : 8
Current Tx epoll                     : 0

Paused by TCP Tx Full                : 0
Resumed by TCP Tx below threshold    : 0
Paused by TCP Buffer Consumed        : 0
Resumed by TCP Buffer Released       : 0
SSL Pause Done                       : 0
SSL Resume Done                      : 0
SNORT Pause Done                     : 0
SNORT Resume Done                    : 0
EV SSL Pause Process                 : 0
EV SNORT Pause Process               : 0
EV SSL/SNORT Resume Process          : 0
Socket Pause Done                    : 0
Socket Resume Done                   : 0
SSL Pause Called                     : 0
SSL Resume Called                    : 0
Async Events Sent                    : 0
Async Events Processed               : 0
```

```
Tx Async Events Sent                : 369
Tx Async Events Recvd               : 369
Tx Async Events Processed           : 369
Failed Send                         : 0

TCP SSL Reset Initiated             : 0
TCP SNORT Reset Initiated           : 0
TCP FIN Received from clnt/svr      : 0
TCP Reset Received from clnt/svr    : 2
SSL FIN Received -> SC              : 0
SSL Reset Received -> SC            : 0
SC FIN Received -> SSL              : 0
SC Reset Received -> SSL            : 0
SSL FIN Received -> TCP             : 0
SSL Reset Received -> TCP           : 0
TCP FIN Processed                   : 0
TCP FIN Ignored FD Already Closed   : 0
TCP Reset Processed                 : 4
SVC Reset Processed                 : 0

Flow Cleaned with Client Data       : 0
Flow Cleaned with Server Data       : 0
Buffers dropped in Tx socket close  : 0
TCP 4k Allocated Buffers            : 369
TCP 16k Allocated Buffers           : 0
TCP 32k Allocated Buffers           : 0
TCP 128k Allocated Buffers          : 0
TCP Freed Buffers                   : 369
SSL Allocated Buffers               : 0
SSL Freed Buffers                   : 0
TCP Received Buffers                : 365
TCP to SSL Enqueued Buffers         : 0
SSL to SVC Enqueued Buffers         : 0
SVC to SSL Enqueued Buffers         : 0
SSL to TCP Enqueued Buffers         : 0
TCP Buffers Sent                    : 365

TCP Failed Buffers Allocations      : 0
TCP Failed 16k Buffers Allocations  : 0
TCP Failed 32k Buffers Allocations  : 0
TCP Failed 128k Buffers Allocations : 0
SSL Failed Buffers Allocations      : 0
Rx Sock Bytes Read < 512            : 335
Rx Sock Bytes Read < 1024           : 25
Rx Sock Bytes Read < 2048           : 5
Rx Sock Bytes Read < 4096           : 0
SSL Server Init                     : 0
Flows Dropped-Snort Gbl Health Yellow        : 0
Flows Dropped-Snort Inst Health Yellow       : 0
Flows Dropped-WCAPI Channel Health Yellow    : 0
Total WCAPI snd flow create svc chain failed : 0
Total WCAPI send data svc chain failed       : 0
Total WCAPI send close svc chain failed      : 0
Total Tx Enqueue Failed                      : 0
Total Cleanup Flow Msg Add to wk_q Failed    : 0
Total Cleanup Flow Msg Added to wk_q         : 0
Total Cleanup Flow Msg Rcvd in wk_q          : 0
Total Cleanup Flow Ignored, Already Done     : 0
Total Cleanup SSL Msg Add to wk_q Failed     : 0
Total UHI mmap                               : 24012
Total UHI munmap                             : 389
```

```
Total Enable Rx Enqueued                    : 0
Total Enable Rx Called                      : 0
Total Enable Rx Process Done                : 0
Total Enable Rx Enqueue Failed              : 0
Total Enable Rx Process Failed              : 0
Total Enable Rx socket on Client Stack Close : 0
Total Enable Rx socket on Server Stack Close : 0

AppQoE_R2#
```

從17.2開始將TCPOpt與其他AppQoE/UTD功能配合使用

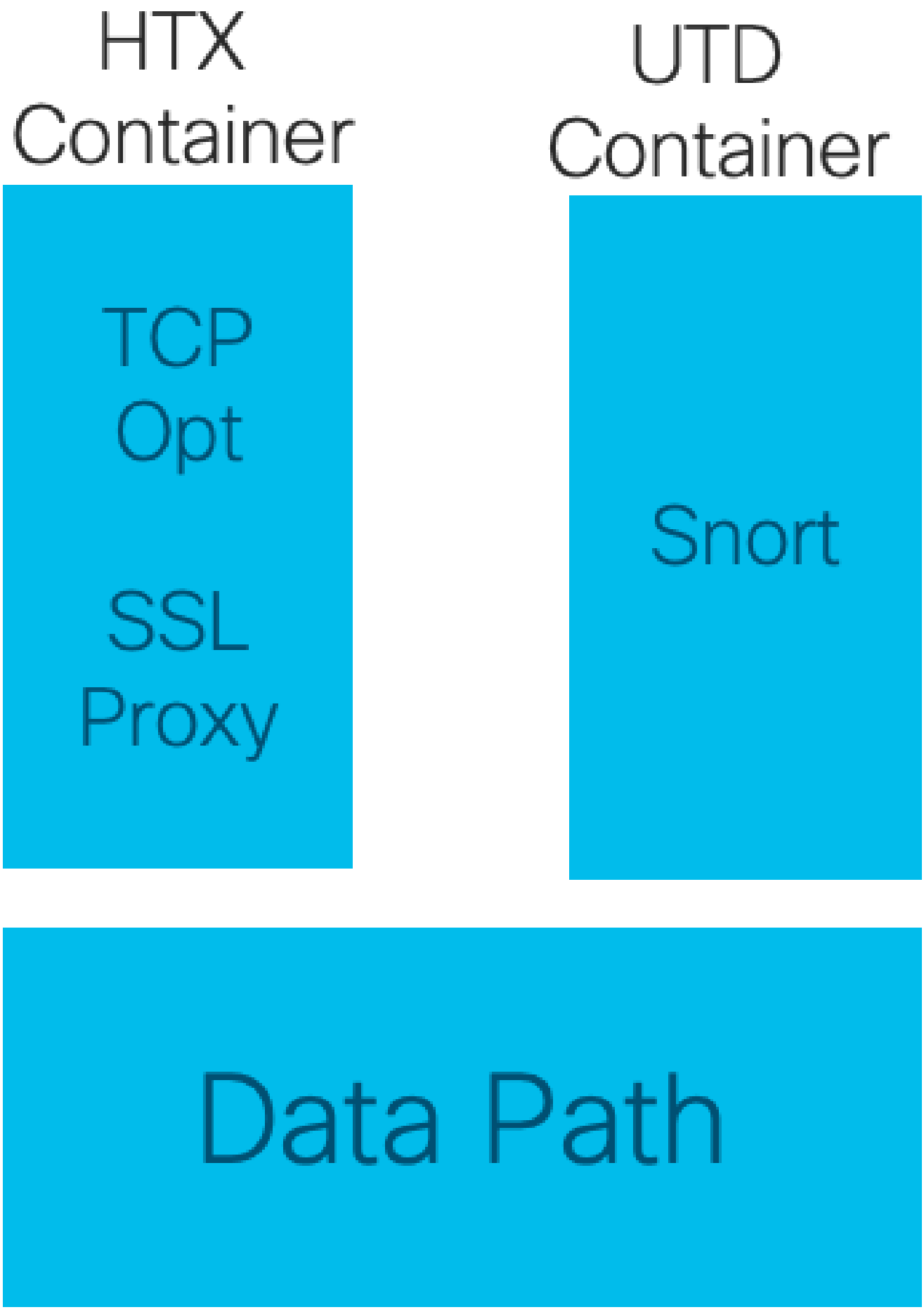
在16.12中，TCPOpt的主要使用情形是Branch-to-Branch。在16.12中有單獨的重新導向到TCP代理，也有單獨的重新導向到UTD容器，這就是為什麼TCP Opt不能與16.12的安全性協同工作

在17.2中實施了一個集中策略路徑，該路徑將檢測對TCP選擇和安全性的需求。

相關資料包將僅重定向到服務平面（點選）一次。

從17.2開始，可能會出現不同的流量選項：

1. 僅TCP選項
2. 僅限UTD
3. TCP選項 — > UTD
4. TCP選擇 — > SSL代理 — > UTD


HTX
Container
UTD
Container
TCP
Opt
SSL
Proxy
Snort
Data Path

相關資訊

- Cisco IOS XE SD-WAN版本16.12.x發行說明
- 技術支援與文件 - Cisco Systems

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。