

Configureren van TCP-optimalisatiefunctie op Cisco IOS® XE SD-WAN cEdge-routers

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Probleem](#)

[Oplossing](#)

[Ondersteunde XE SD-WAN platforms](#)

[Voorbehouden](#)

[Configureren](#)

[Use Case 1. Configureer TCP-optimalisatie op een branch \(alles in één cEdge\)](#)

[Use Case 2. Configureer TCP-optimalisatie in datacenter met een extern SN](#)

[failover-case](#)

[Verifiëren](#)

[Problemen oplossen](#)

[Gerelateerde informatie](#)

Inleiding

Dit document beschrijft de Optimalisatiefunctie voor Transmission Control Protocol (TCP) op Cisco IOS® XE SD-WAN routers, die in augustus 2019 werd geïntroduceerd in versie 16.12. De behandelde onderwerpen zijn vereisten, probleembeschrijving, oplossing, de verschillen in TCP-optimalisatiealgoritmen tussen Viptela OS (vEdge) en XE SD-WAN (cEdge), configuratie, verificatie en lijst van gerelateerde documenten.

Voorwaarden

Vereisten

Er zijn geen specifieke vereisten van toepassing op dit document.

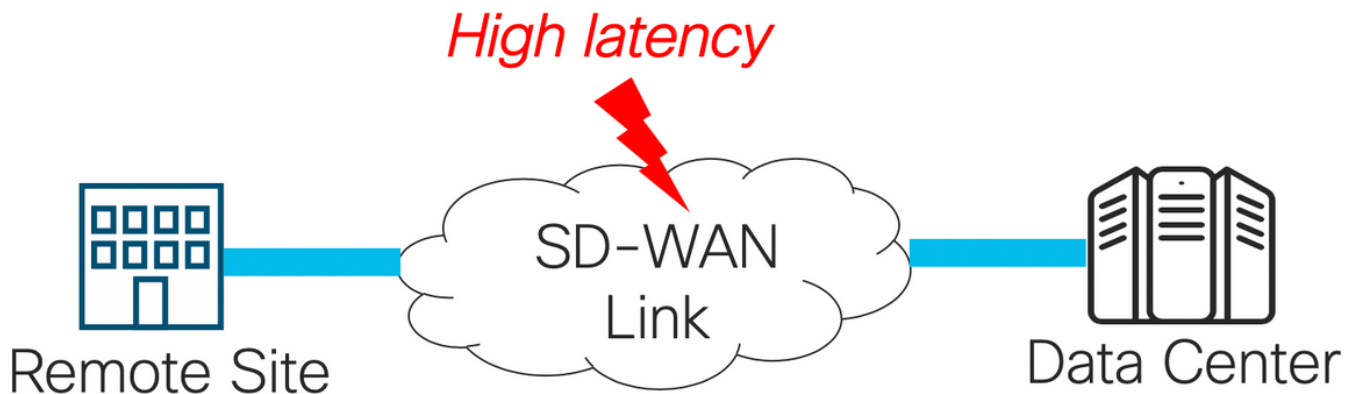
Gebruikte componenten

De informatie in dit document is gebaseerd op Cisco IOS® XE SD-WAN.

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

Probleem

Hoge latentie op een WAN-verbinding tussen twee SD-WAN-kanten veroorzaakt slechte toepassingsprestaties. U hebt kritisch TCP-verkeer, dat moet worden geoptimaliseerd.



Oplossing

Wanneer u de functie voor TCP-optimalisatie gebruikt, verbetert u de gemiddelde TCP-doorvoersnelheid voor kritische TCP-stromen tussen twee SD-WAN-sites.

Bekijk het overzicht en de verschillen tussen TCP-optimalisatie op cEdge Bottleneck Bandwidth en Round-trip (BBR) en vEdge (CUBIC)

Het snelle BBR-propagatietijdalgoritme wordt gebruikt in de XE SD-WAN implementatie (op cEdge).

Viptela OS (vEdge) heeft een ander, ouder algoritme, genaamd CUBIC.

CUBIC houdt voornamelijk rekening met pakketverlies en wordt op grote schaal geïmplementeerd op verschillende client-besturingssystemen. Windows, Linux, MacOS en Android hebben CUBIC al ingebouwd. In sommige gevallen, waar u oude clients met TCP stack zonder CUBIC, waardoor TCP optimalisatie op vEdge mogelijk maakt, brengt verbeteringen. Een van de voorbeelden waar vEdge TCP/CUBIC-optimalisatie heeft geprofiteerd, is op onderzeeërs die oude clienthosts en WAN-links gebruiken die aanzienlijke vertragingen/dalingen ondervinden. Let op dat alleen vEdge 1000 en vEdge 2000 TCP/CUBIC ondersteunen.

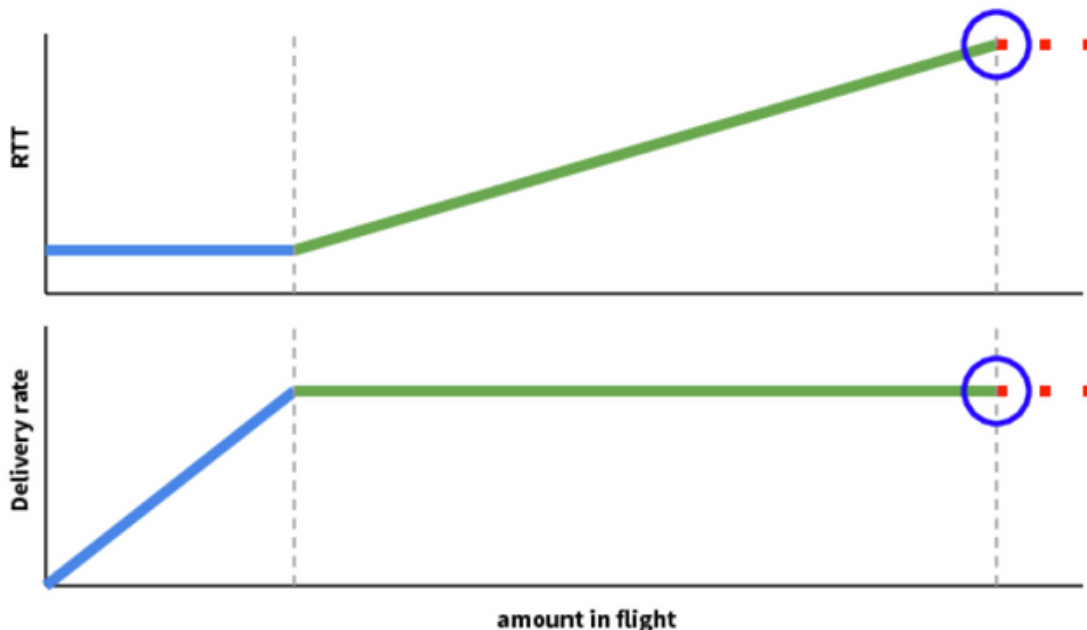
BBR is vooral gericht op ronde-trip tijd en latentie. Geen pakketverlies. Als je pakketten van US West naar East Coast of zelfs naar Europa stuurt via het openbare internet, zie je in de meeste gevallen geen pakketverlies. Het openbaar internet is soms te goed in termen van pakketverlies. Maar wat je ziet is vertraging/latentie. Dit probleem wordt aangepakt door BBR, dat in 2016 door Google werd ontwikkeld.

In een notendop, BBR modelleert het netwerk en bekijkt elke erkenning (ACK) en updates max Bandbreedte (BW) en minimum Ronde Reistijd (RTT). Dan controle het verzenden is gebaseerd op model: sonde voor max BW en min RTT, tempo dichtbij schatting BW en houden inflight dichtbij Bandwidth-Delay-Product (BDP). Het belangrijkste doel is om een hoge doorvoersnelheid te garanderen met een kleine knelpuntwachtrij.

Op deze dia van [Mark Claypool](#) staat het gebied waar CUBIC werkt:

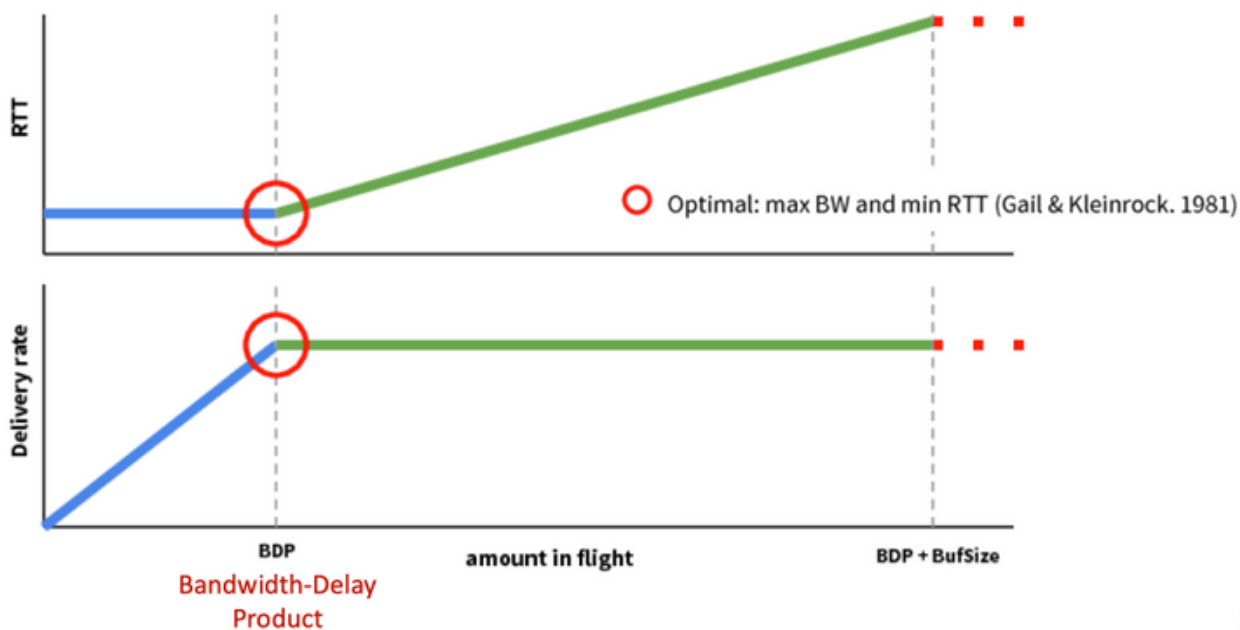
Congestion and Bottlenecks

○ CUBIC / Reno



BBR werkt op een betere plaats, zoals ook te zien is op deze dia van Mark Claypool:

Congestion and Bottlenecks



Als u meer over het BBR-algoritme wilt lezen, vindt u verschillende publicaties over BBR die zijn gekoppeld bovenaan de bbr-dev mailinglijst startpagina [Hier](#).

Samenvattend:

Platform en algoritme	Belangrijke invoerparameter	Use case
cEdge (XE SD-WAN): BBR	RTT/Latency	Kritisch TCP-verkeer tussen twee WAN-sites
vEdge (Viptela OS): KUBICP	pakketverlies	Oude clients zonder TCP-optimalisatie

Ondersteunde XE SD-WAN platforms

In de XE SD-WAN SW release 16.12.1d ondersteunen deze cEdge-platforms TCP-optimalisatie BBR:

- ISR-4331
- ISR 4351 router
- CSR1000v met 8 vCPU's en min. 8 GB RAM

Voorbehouden

- Alle platforms met DRAM van minder dan 8 GB RAM worden momenteel niet ondersteund.
- Alle platforms met vier of minder datakernen worden momenteel niet ondersteund.
- TCP-optimalisatie ondersteunt MTU 2000 niet.
- Momenteel - geen ondersteuning voor IPv6-verkeer.
- Optimalisatie voor DIA-verkeer naar een BBR-server van derden wordt niet ondersteund. Je hebt aan beide kanten een cEdge SD-WAN router nodig.
- In het datacenterscenario wordt momenteel slechts één Service Node (SN) ondersteund per één Control Node (CN).
- Op dit moment wordt een gecombineerde toepassing met security (UTD-container) en TCP-optimalisatie op hetzelfde apparaat niet ondersteund.

Opmerking: ASR1k ondersteunt momenteel geen TCP-optimalisatie. Er is echter een oplossing voor ASR1k, waarbij de ASR1k TCP-verkeer via AppNav-tunnel (GRE ingekapseld) naar een externe CSR1kv stuurt voor optimalisatie. Momenteel (februari 2020) wordt slechts één CSR1k als extern serviceknooppunt ondersteund. Dit wordt later beschreven in het configuratiegedeelte.

In deze tabel worden de voorbehouden per release samengevat en worden de ondersteunde hardwareplatforms onderstreept:

Scenario's	Use cases	16.12.1	17.2.1	17.3.1	17.4.1	Opmerkingen
Vestigings-aan-internet	DIA	Nee	Ja	Ja	Ja	In 16.12.1 wordt AppNav-tunnel niet ingeschakeld op de internetinterface
	SAAS	Nee	Ja	Ja	Ja	In 16.12.1 wordt AppNav-tunnel niet ingeschakeld op de internetinterface
	Single Edge-router	Nee	Nee	EFT	Ja	Ondersteuning van meerdere SN nodig Behoeft aan stroomsymmetrie of AppNav-tunnel
Vestigings-aan-DC	Meervoudige Edge-routers	Nee	Nee	EFT	Ja	AppNav-tunnel stroomsynchronisatie 16.12.1 niet getest met meerdere SN
	Meervoudige SN's	Nee	Nee	EFT	Ja	Verbetering in vMan om meerdere SN IP's te accepteren
Vestigings-aan-vertakking	Volledig mesh-netwerk (Gesproken)	Ja	Ja	Ja	Ja	

	hub and Spoke (Spoke-Hub- Spoke)	Nee	Ja	Ja	Ja
Ondersteuning van BBR	TCP kiezen met BBR	gedeeltelij k	gedeeltelij k	Full	Full
Platforms	Ondersteunde platforms	Alleen 4300 en CSR	Alles, behalve ISR1100	Alle	Alle

Configureren

Een concept van SN en CN wordt gebruikt voor TCP-optimalisatie:

- SN is een daemon, die verantwoordelijk is voor de daadwerkelijke optimalisatie van TCP-stromen.
- CN staat bekend als AppNav Controller en is verantwoordelijk voor verkeersselectie en transport van/naar SN.

SN en CN kunnen op dezelfde router worden uitgevoerd of als verschillende knooppunten worden gescheiden.

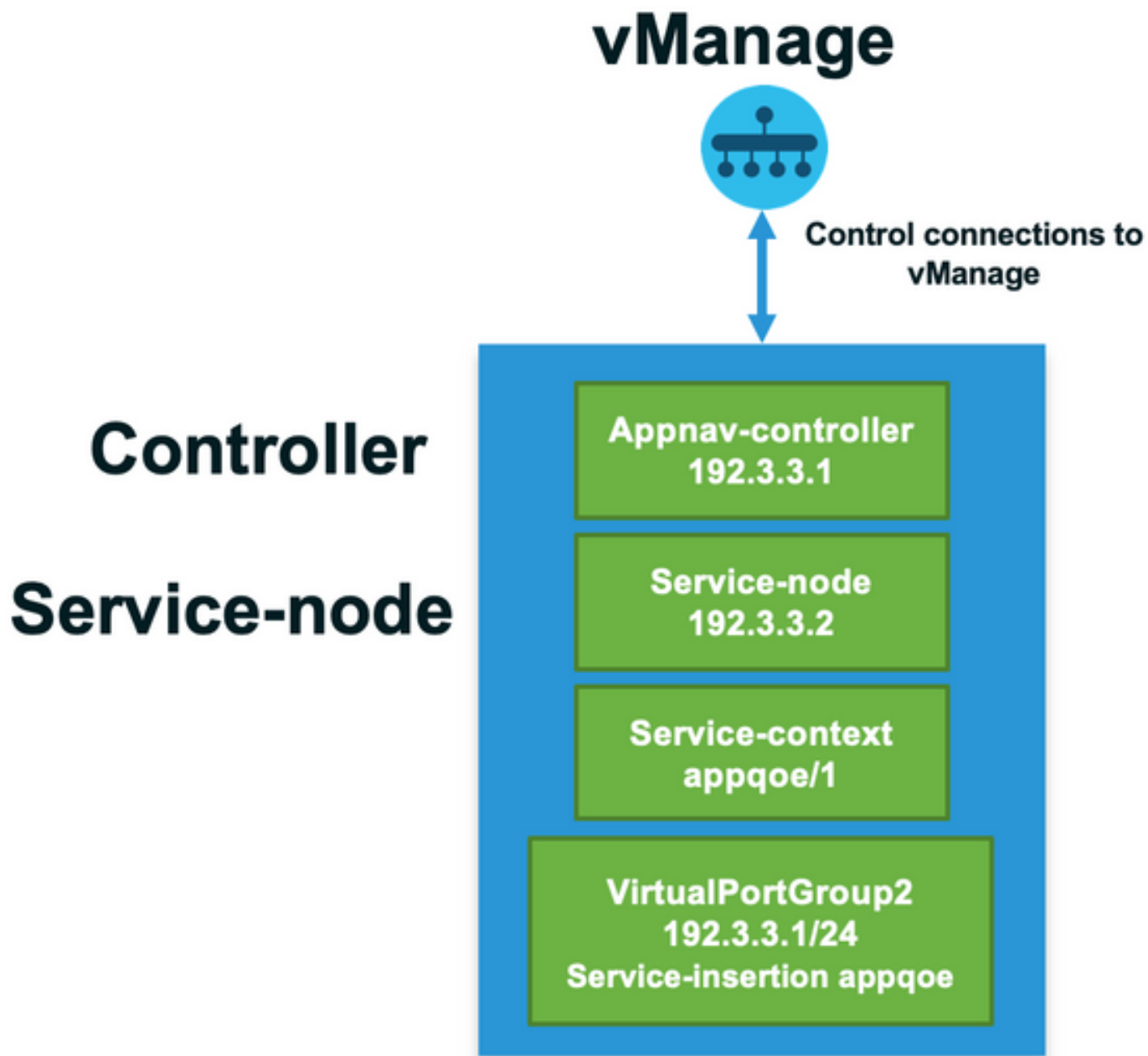
Er zijn twee belangrijke gevallen van gebruik:

1. Verdeel het gebruiksgeschied met SN en CN die op de zelfde router ISR4k lopen.
2. Data Center use case, waarbij CN draait op ASR1k en SN draait op een aparte CSR1000v virtuele router.

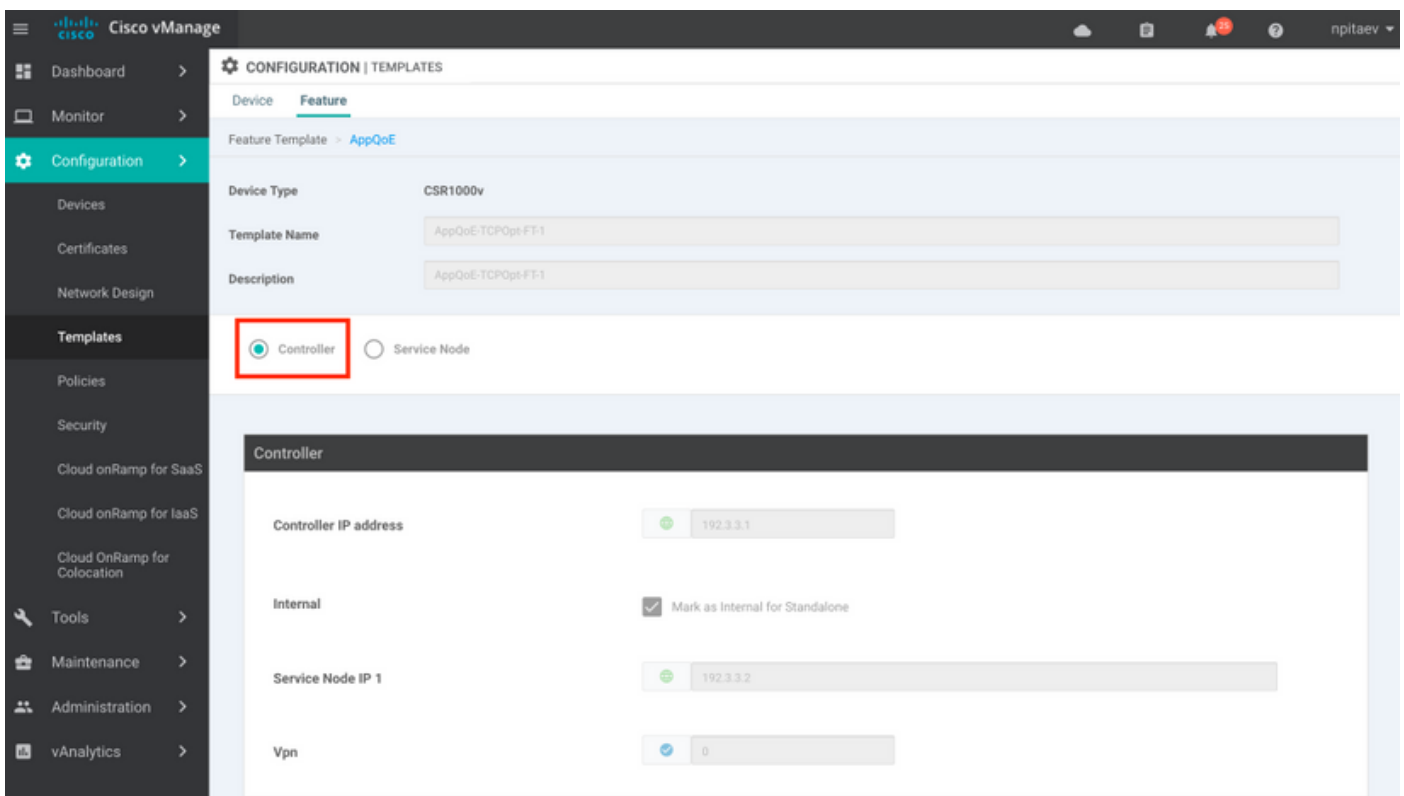
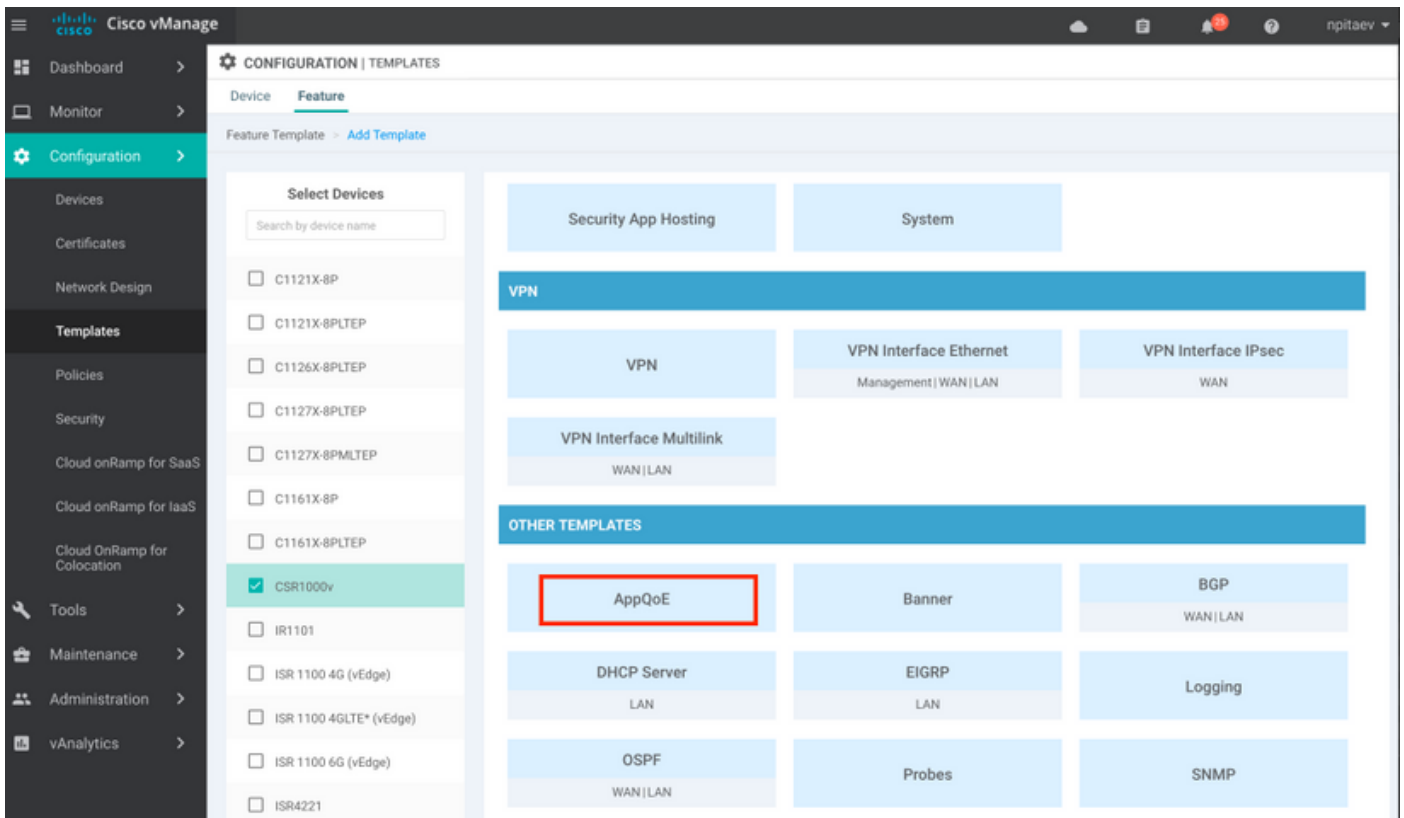
Beide gevallen van gebruik worden in deze rubriek beschreven.

Use Case 1. Configureer TCP-optimalisatie op een branch (alles in één cEdge)

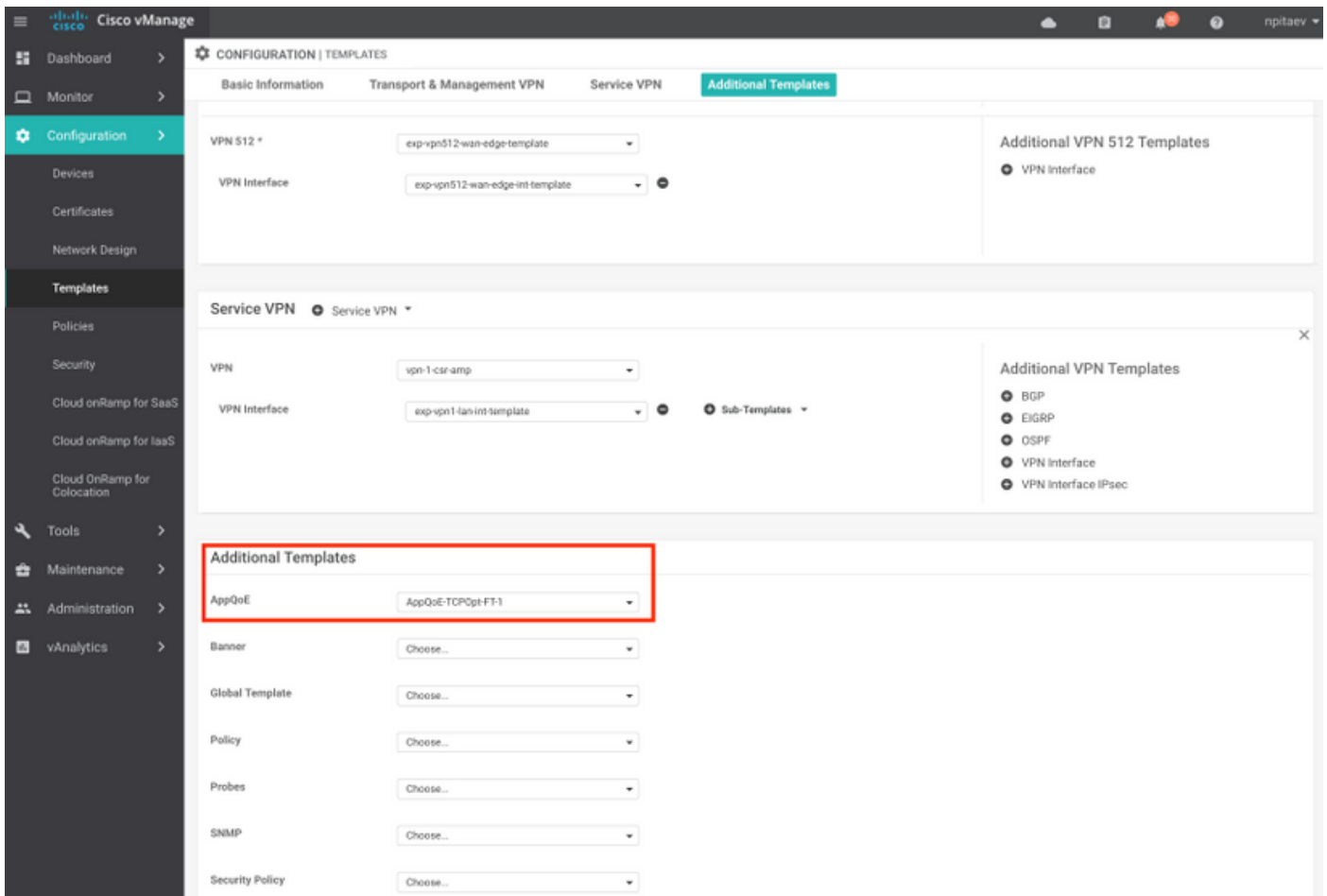
Dit beeld toont de algemene interne architectuur voor één enkele standalone optie bij een tak:



Stap 1. Om TCP-optimalisatie te kunnen configureren, moet u een functiesjabloon voor TCP-optimalisatie in vManager maken. Navigeer naar **Configuration > Templates > Feature Templates > Andere sjablonen > AppQoE** zoals in de afbeelding.



Stap 2. Voeg de AppQoE-functiesjabloon toe aan de juiste apparaatsjabloon onder **Aanvullende sjablonen**:



Hier is de CLI-voorvertoning van de Sjabloonconfiguratie:

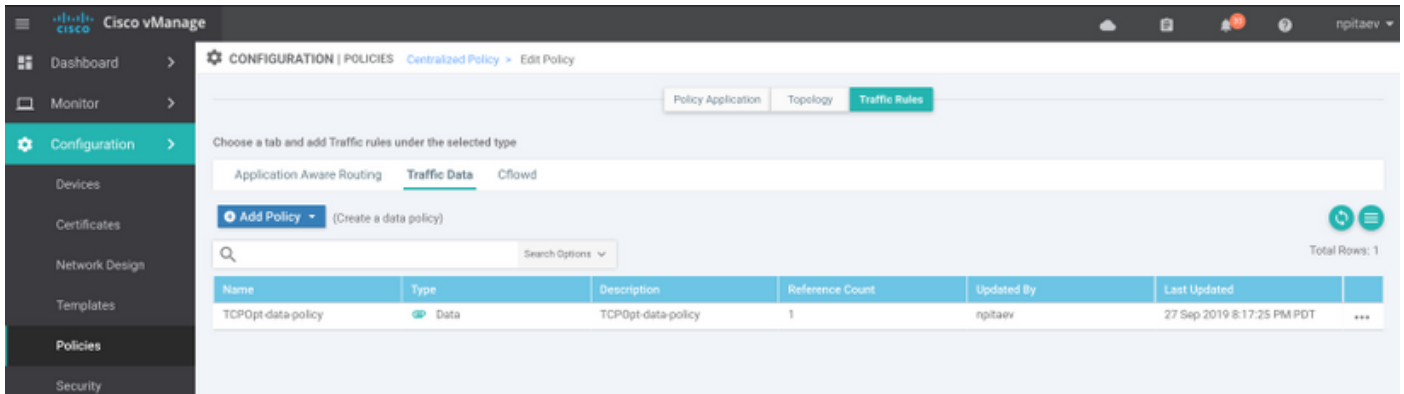
```

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
!

```

Stap 3. Maak een gecentraliseerd gegevensbeleid met de definitie van het interessante TCP-verkeer voor optimalisatie.

Als voorbeeld; dit gegevensbeleid past IP prefix 10.0.0.0/8, dat bron en bestemmingsadressen omvat, aan en laat de optimalisering van TCP voor het toe:



Hier is de CLI preview van het vSmart Policy:

```

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
!
!

```

Use Case 2. Configureer TCP-optimalisatie in datacenter met een extern SN

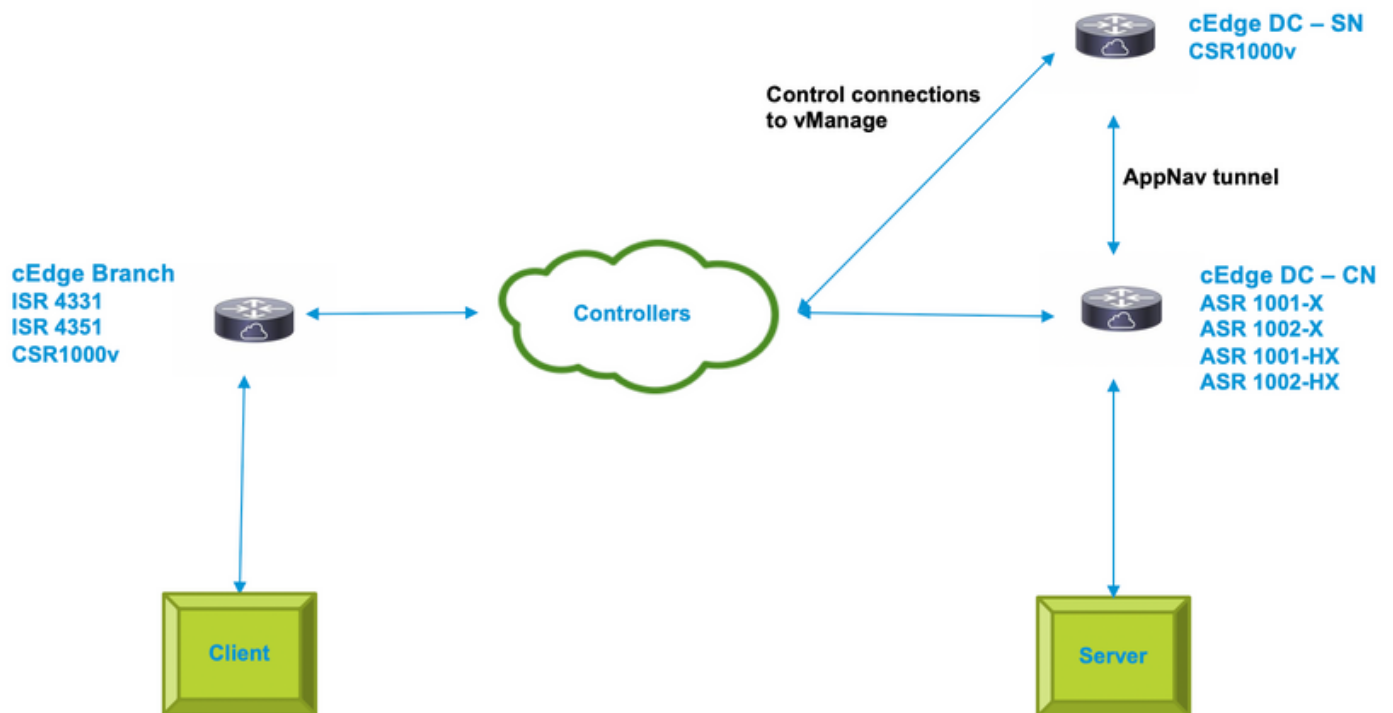
Het belangrijkste verschil met het geval van branchegebruik is de fysieke scheiding van SN en CN. In de all-in-one branch use case wordt de connectiviteit gedaan binnen dezelfde router met

behelp van Virtual Port Group Interface. In het datacentergebruiksgeval is er een AppNav GRE- ingekapselde tunnel tussen ASR1k als CN en externe CSR1k als SN. Er is geen behoefte aan een specifieke link of cross-connect tussen CN en externe SN, eenvoudige IP bereikbaarheid is genoeg.

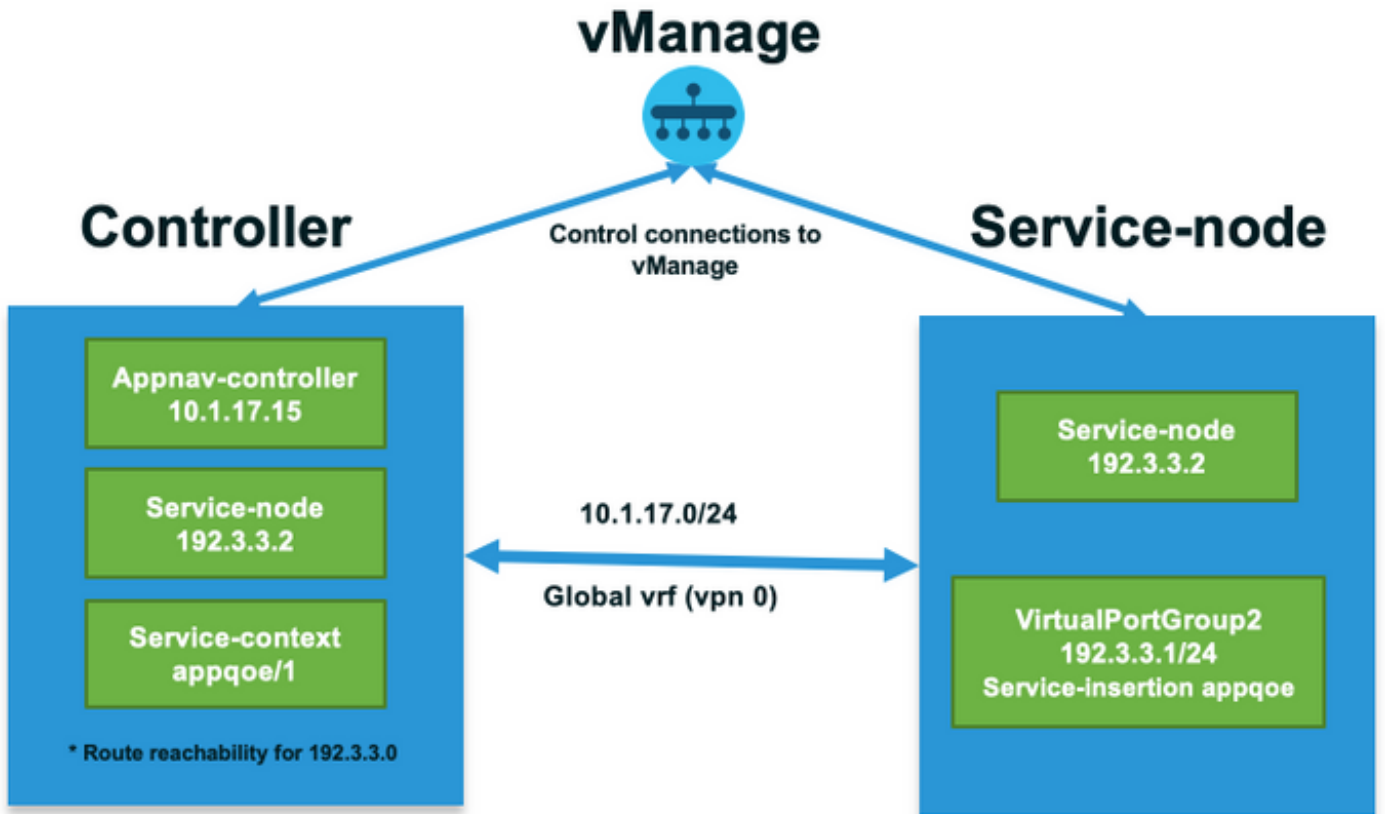
Er is één AppNav (GRE) tunnel per SN. Voor toekomstig gebruik, waar meerdere SNs worden ondersteund, is het aanbevolen om /28 subnetverbinding te gebruiken voor het netwerk tussen CN en SN.

Twee NIC's worden aanbevolen voor een CSR1k die fungeert als SN. 2e NIC voor SD-WAN controller is nodig als SN moet worden geconfigureerd / beheerd door vManager. Als SN handmatig wordt geconfigureerd/beheerd, is 2e NIC optioneel.

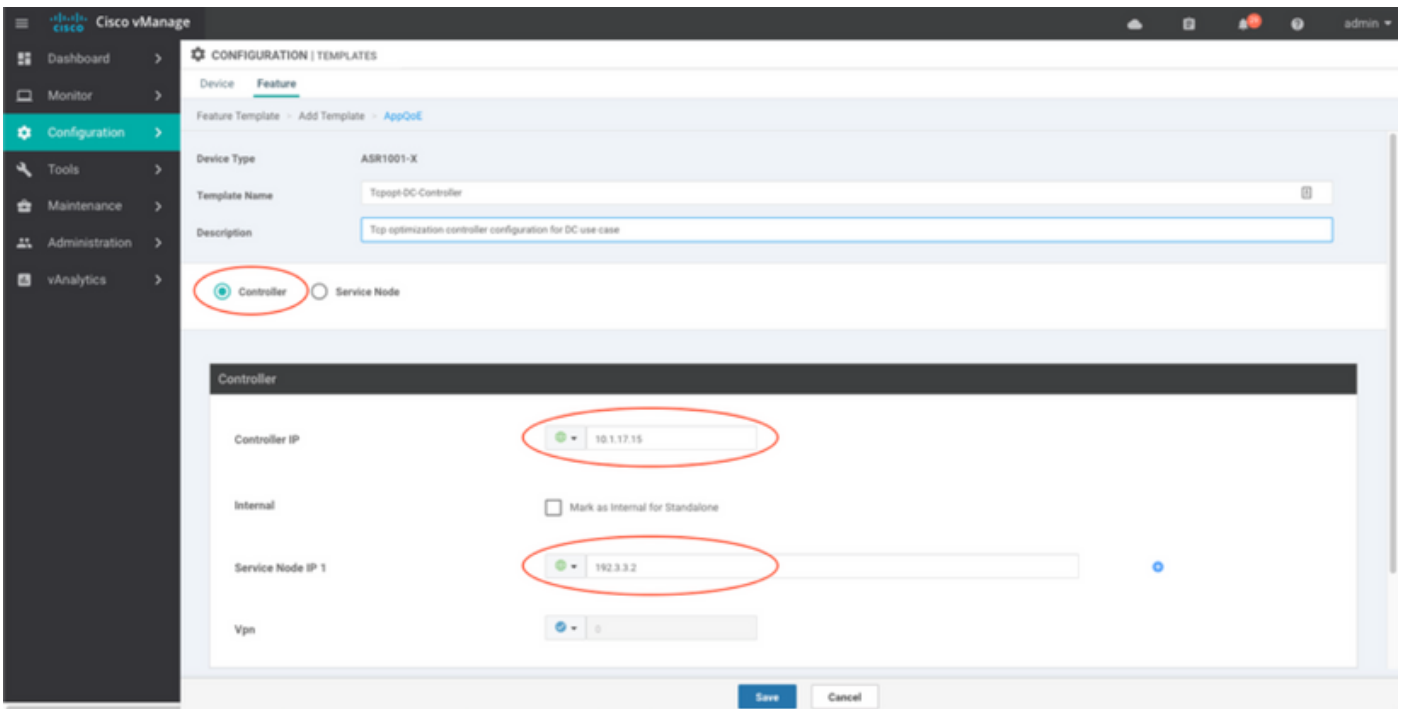
Dit beeld toont datacenter ASR1k dat draait als CN en CSR1kv als Service Node SN:



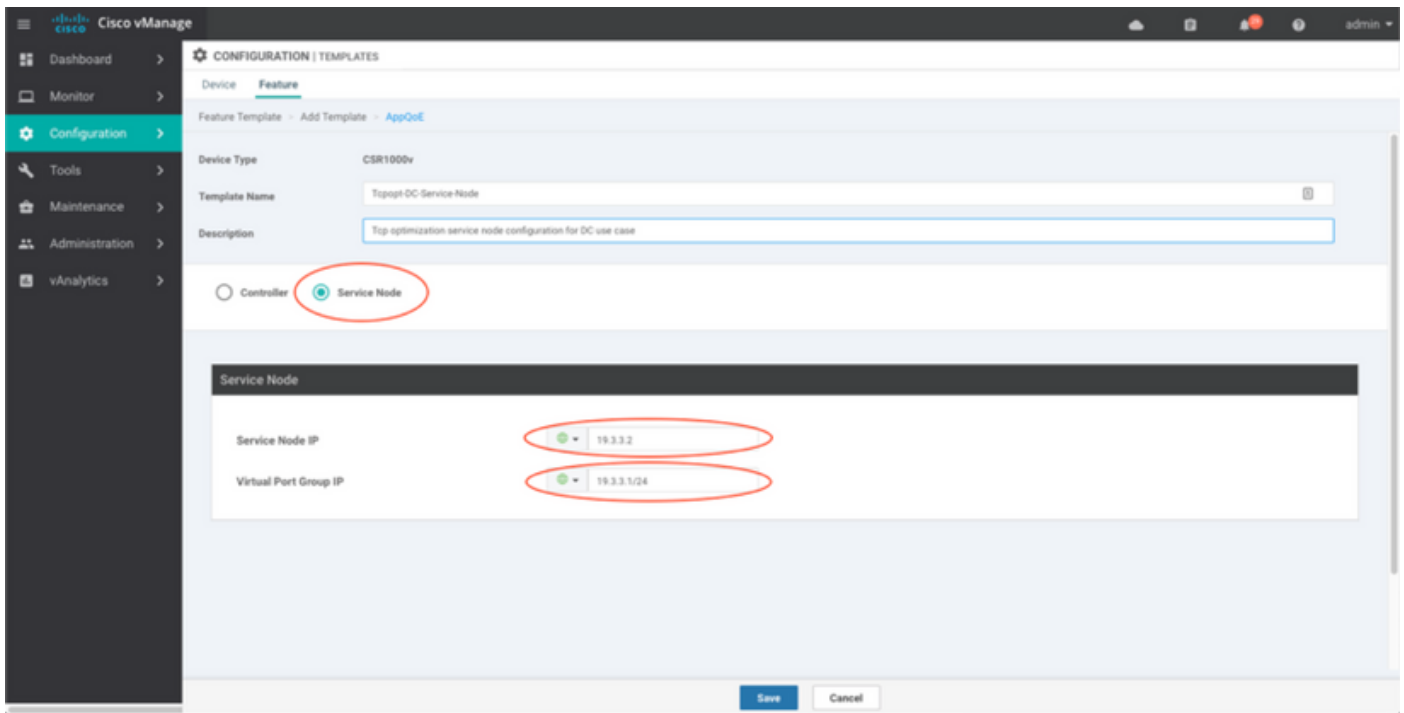
De topologie voor de datacentergebruikscase met ASR1k en extern CSR1k wordt hier weergegeven:



Deze AppQoE-functiesjabloon toont ASR1k geconfigureerd als controller:



CSR1k geconfigureerd als extern serviceknooppunt wordt hier weergegeven:



failover-case

failover in de datacentergebruikscase met CSR1k als SN, in het geval van een externe CSR1k-storing:

- TCP-sessies die al bestaan, gaan verloren omdat de TCP-sessie op SN is beëindigd.
- Nieuwe TCP-sessies worden naar de eindbestemming verzonden, maar TCP-verkeer wordt niet geoptimaliseerd (omzeilen).
- Geen versperring voor interessant verkeer in het geval van SN falen.

De detectie van failover is gebaseerd op AppNav-hartslag, die 1 slag per seconde is. Na 3 of 4 fouten, wordt de tunnel zoals beneden verklaard.

failover in de branch use case is vergelijkbaar - in het geval van SN-fout wordt het verkeer niet-geoptimaliseerd rechtstreeks naar de bestemming gestuurd.

Verifiëren

Gebruik deze sectie om te controleren of uw configuratie goed werkt.

Controleer TCP-optimalisatie op CLI met het gebruik van deze CLI-opdracht en zie de samenvatting van de geoptimaliseerde stromen:

```
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#
```

Deze output geeft gedetailleerde informatie over geoptimaliseerde stromen:

```
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
```

```
-----
appgoe: flow action DIVERT, svc_idx 0, divert pkt_cnt 1, bypass pkt_cnt 0, drop pkt_cnt 0,
inject pkt_cnt 1, error pkt_cnt 0, ingress_intf Tunnel2, egress_intf GigabitEthernet3
=====
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
```

```
-----
appgoe: flow action DIVERT, svc_idx 0, divert pkt_cnt 158, bypass pkt_cnt 0, drop pkt_cnt 0,
inject pkt_cnt 243, error pkt_cnt 0, ingress_intf Tunnel2, egress_intf GigabitEthernet3
=====
+++++
Number of flows that passed filter: 4
+++++
          FLOWS DUMP DONE.
+++++
```

```
BR11-CSR1k#
```

Problemen oplossen

Er is momenteel geen specifieke troubleshooting-informatie beschikbaar voor deze configuratie.

Gerelateerde informatie

- [Releaseopmerkingen voor Cisco IOS XE SD-WAN release 16.12.x](#)
- [Cisco SD-WAN releases 19.1, 19.2 - TCP-optimalisatiegids configureren](#)
- [Cisco SD-WAN configureert TCP-optimalisatie voor vEdge](#)
- [Technische ondersteuning en documentatie – Cisco Systems](#)

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document ([link](#)) te raadplegen.