

QoS-spraakproblemen oplossen

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Conventies](#)

[Oorzaken van Choppy Voice](#)

[Bandbreedtevereisten](#)

[Prioriteit spraakverkeer](#)

[Servische vertraging](#)

[VAD](#)

[Typische Configuratievoorbeelden voor QoS](#)

[Mechanisme voor jitter en playout](#)

[Indelingsmechanisme](#)

[Jitter Buffer](#)

[Timer en jitter identificeren](#)

[roep actieve stem tonen](#)

[Bel <port-number>](#)

[Jitter Buffer op een gateway configureren](#)

[Indelingsmodus](#)

[Gerelateerde informatie](#)

[Inleiding](#)

Om Packet Voice een realistische vervanging te zijn voor de standaardtelefoniediensten van het openbare telefoonnetwerk (PSTN), moet de ontvangen kwaliteit van Packet Voice vergelijkbaar zijn met die van basistelefoonservices. Dit betekent constant spraakuitzendingen van hoge kwaliteit. Zoals andere real-time toepassingen heeft Packet Voice een brede bandbreedte en is de vertraginggevoeligheid. Om spraaktransmissies begrijpelijk (niet veranderlijk) naar de ontvanger te laten verlopen, kunnen spraakpakketten niet worden gedropt, te lang worden uitgesteld of lijden verschillende vertraging (ook wel bekend als jitter). Dit document beschrijft verschillende QoS-overwegingen (Quality of Service) die probleemoplossing ondersteunen bij het oplossen van problemen met spraakproblemen. De belangrijkste redenen voor kokerproblemen zijn verloren en vertraagde spraakpakketten.

[Voorwaarden](#)

[Vereisten](#)

Lezers van dit document zouden op de hoogte moeten zijn van:

- Basisconfiguratie van Packet Voice (VoIP, Voice over Frame Relay (VoFR) of Voice over ATM (VoATM) volgens hun vereisten).
- Basisbegrip van spraakprioritering, fragmentatie, verschillende codecs en hun bandbreedtevereisten.

Gebruikte componenten

De informatie in dit document is van toepassing op alle Cisco-spraakgateways software en hardwareversies.

De informatie in dit document is gebaseerd op apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als u in een levend netwerk werkt, zorg er dan voor dat u de potentiële impact van om het even welke opdracht begrijpt alvorens het te gebruiken.

Conventies

Raadpleeg [Cisco Technical Tips Conventions](#) (Conventies voor technische tips van Cisco) voor meer informatie over documentconventies.

Oorzaken van Choppy Voice

De verandering van stemkwaliteit wordt veroorzaakt door spraakpakketten die of variabel uitgesteld of in het netwerk verloren worden. Wanneer een stempakket vertraagd is bij het bereiken van zijn bestemming, heeft de doelgateway een verlies van real-time informatie. In deze gebeurtenis, moet de bestemming gateway voorspellen wat de inhoud van het gemiste pakket kan zijn. De voorspelling leidt ertoe dat de ontvangen stem niet dezelfde kenmerken heeft als de uitgezonden stem. Dit leidt tot een ontvangen stem die robotisch klinkt. Als een spraakpakket wordt uitgesteld buiten de voorspellingscapaciteit van een ontvangende gateway, blijft de real-time kloof leeg. Met niets om dat gat op te vullen aan het ontvangende eind, gaat een deel van de uitgezonden toespraak verloren. Dit resulteert in een hachelijke stem. Veel spraakproblemen worden opgelost door ervoor te zorgen dat de spraakpakketten niet erg lang worden gewacht (en zelfs nog meer, niet wisselend worden uitgesteld). Soms voegt VAD (Voice Interface Detectie) front-end knippering toe aan een spraakgesprek. Dit is een andere oorzaak van hachelijke (of geknipte) stem.

De verschillende secties in dit document tonen hoe u het geval van hachelijke stemmen kunt minimaliseren. De meeste van deze maatregelen vereisen de introductie van een minimum jitter in uw spraaknetwerk.

Bandbreedtevereisten

Voordat u overweegt om maatregelen toe te passen om jitter te minimaliseren, voorzien voldoende netwerkbandbreedte om spraakverkeer in real-time te ondersteunen. Zo klinkt een 80 kbps G.711 VoIP-oproep (64 kbps payload + 16 kbps header) naast een 64 kbps link omdat minstens 16 kbps van de pakketten (wat 20 procent is) gevallen zijn. De bandbreedtevereisten variëren op basis van de codec die voor compressie wordt gebruikt. Verschillende codecs hebben verschillende payload- en headereisen. Het gebruik van VAD beïnvloedt ook het

bandbreedtevereiste. Als Real Time Protocol (RTP) headercompressie (cRTP) wordt gebruikt, kan dit het bandbreedte-vereiste verder verlagen.

Bijvoorbeeld, de bandbreedte die voor een stemvraag wordt vereist het gebruiken van de codec G.729 (standaard 20 byte payload) met cRTP, is zoals deze:

- Voice payload + gecompriemd (RTP-header + User Datagram Protocol (UDP)-header + IP-kop) + Layer 2 header

Dit staat gelijk aan:

- 20 bytes + gecompriemd (12 bytes + 8 bytes + 20 bytes) + 4 bytes

Dit is gelijk aan:

- 28 bytes, aangezien de headercompressie de IP RTP header reduceert tot maximaal 4 bytes. Dit levert een snelheid van 11,2 kbps op met een snelheid van 8 kbps (50 pakketten per seconde).

Raadpleeg voor meer informatie [Voice-over-IP-band per gespreksbandbreedte](#).

Prioriteit spraakverkeer

Er zijn twee belangrijke componenten in het prioriteren van stem. Het eerste is het classificeren en markeren van interessant spraakverkeer. Het tweede is het prioriteren van het gemarkeerde interessante spraakverkeer. De twee subsecties hier bespreken verschillende benaderingen van het classificeren, markeren en prioriteren van stem.

Classificatie en markering

Om bandbreedte voor VoIP-pakketten te garanderen, moet een netwerkkapparaat de pakketten in al het IP-verkeer kunnen identificeren dat er door stroomt. Netwerkkapparaten gebruiken het bron- en doeladres in de IP-header, of de bron- en doelnummers in de UDP-header om VoIP-pakketten te identificeren. Dit identificatie- en groeperingsproces wordt classificatie genoemd. Het vormt de basis voor het verstrekken van QoS.

Packet classificatie kan processor-intensief zijn. Daarom moet de classificatie zoveel mogelijk naar de rand van het netwerk worden gericht. Omdat elke hop nog steeds een bepaling moet maken over de behandeling die een pakje zou moeten ontvangen, moet u een eenvoudiger, efficiëntere classificatiemethode in de netwerkkern hebben. Deze eenvoudiger classificatie wordt bereikt door het markeren of instellen van de Type of Service (ToS)-byte in de IP-header. De drie belangrijkste bits van de ToS-byte worden IP-prioriteitsbits genoemd. De meeste toepassingen en verkopers ondersteunen momenteel het instellen en herkennen van deze drie bits. Marking evolueert zodat de zes meest significante bits van de ToS-byte, het gedifferentieerde servicescopoint (DSCP) genoemd, kunnen worden gebruikt. Raadpleeg het verzoek om opmerkingen (RFC).

Gedifferentieerde services (DiffServ) is een nieuw model waarin verkeer wordt behandeld door intermediaire systemen met relatieve prioriteiten gebaseerd op het ToS-veld. De [DiffServ-standaard](#), gedefinieerd in [RFC 2474](#) en [RFC 2475](#), vervangt de oorspronkelijke specificatie voor het definiëren van de pakketprioriteit die in [RFC 791](#) is beschreven. DiffServ verhoogt het aantal definieerbare prioriteitsniveaus door bits van een IP-pakket opnieuw toe te wijzen voor prioriteitsmarkering. De DiffServ-architectuur definieert het DiffServ-veld. Het vervangt de ToS-byte in IP V4 om besluiten per-Hop Behavior (PHB) te maken over pakketclassificatie en traffic

shaping-functies zoals meting, markering, vormgeving en toezicht. Naast de eerder genoemde RFC's, definieert [RFC 2597](#) de klassen Gegarandeerd doorsturen (AF). Dit is een uitsplitsing van de DSCP-velden. Raadpleeg voor meer informatie over DSCP het [Quality-of-Service beleid met DSCP implementeren](#).

ToS Byte - P2 P1 p0 T3 T1 T0 CU

IP-voorrang: drie bits (P2-P0), ToS: vier bits (T3-T0), CU: stukje

DiffServ-veld - DS5 DS4 DS3 DS2 DS1 DS0-N

DSCP: zes bits (DS5-DS0), ECN: twee bits

XXX00000-bits 0, 1, 2 (DS5, DS4, DS3) zijn prioriteitsbits, waarin:

- 111 = Netwerkcontrole = voorrang 7
- 110 = Internetwork Control = Priority 6
- 101 = KRITIEK/ECP = Voorzorgsmaatregelen 5
- 100 = Flitser Override = Voorrang 4
- 011 = Flitser = Voorrang 3
- 010 = onmiddellijk = voorrang 2
- 001 = Prioriteit = voorrang 1
- 000 = Routine = Voorzorgsmaatregelen 0

000XX00-bits 3, 4, 5 (DS2, DS1, DS0) zijn vertragingbits, doorvoersnelheid en betrouwbaarheid.

- bit 3 = vertraging [D] (0 = normaal; 1 = Laag)
- bit 4 = doorvoersnelheid [T] (0 = normaal; 1 = Hoog)
- bit 5 = betrouwbaarheid [R] (0 = normaal; 1 = Hoog)

00000XX-bits 6, 7: ECN

In deze twee delen wordt gesproken over twee manieren waarop classificatie en markering plaatsvinden.

Spraakkiepeers om pakketten in te delen en te markeren

Met Cisco VoIP gateways, gebruikt u gewoonlijk spraakkiepeers om de VoIP-pakketten aan te geven en de IP-prioriteitsbits te markeren. Deze configuratie toont hoe de IP-prioriteitsbits moeten worden gemarkeerd:

```
dial-peer voice 100 voip
destination-pattern 100
session target ipv4:10.10.10.2
ip precedence 5
```

In het bovenstaande voorbeeld heeft elke VoIP-oproep die overeenkomt met de **stem-peer stem-100 voip**-opdracht alle spraakpakketten ingesteld met IP voorrang 5. Dit betekent dat de drie belangrijkste bits van de IP ToS-byte op 101 zijn ingesteld.

```
dial-peer voice 100 voip
destination-pattern 100
session target ipv4:10.10.10.2
```

```
ip qos dscp ef media
ip qos dscp af31 signaling
```

In het bovenstaande voorbeeld heeft elke VoIP-oproep die overeenkomt met de opdracht **dial-peers spraak 100 voip** alle media payload-pakketten (spraakpakketten) ingesteld met het patroon **FastForwarding (EF) bit** -patroon 10110. Alle signaleringspakketten zijn ingesteld met AF-bit-bit-patroon 0110100.

Opmerking: De **ip qos dscp**-opdracht wordt ondersteund sinds Cisco IOS® softwarerelease 12.2(2)T. Het IP-voorrang is niet langer beschikbaar in Cisco IOS-softwarerelease 12.2T. Hetzelfde wordt echter bereikt met de opdracht **ip qos dscp**. IP-prioriteitskaarten (101) naar IP DSCP 101000. Raadpleeg voor meer informatie de [classificatie van VoIP-signalering en media met DSCP voor QoS](#).

Modulair QoS CLI om pakketten te classificeren en te markeren

De te gebruiken aanbevolen classificatie- en markeringsmethode is de modulaire QoS CLI. Dit is een op een sjabloon gebaseerde configuratiemethode die de classificatie van het beleid scheidt. Dit maakt het mogelijk dat meerdere QoS-functies samen worden geconfigureerd voor meerdere klassen. Gebruik een opdracht **class-map** om verkeer in te delen op basis van verschillende matchcriteria en een opdracht **voor** beleidskaarten om te bepalen wat er met elke klasse moet gebeuren. Pas het beleid op inkomend of uitgaand verkeer op een interface toe door de **dienst-beleid** opdracht uit te geven. Dit configuratievoorbeeld toont hoe u modulaire QoS CLI kunt gebruiken om pakketten te classificeren en te markeren:

```
access-list 100 permit udp any any range 16384 32767
access-list 101 permit tcp any any eq 1720
!
class-map match-all voip
match access-group 100
class-map match-all control
match access-group 101
!
policy-map mqc
class voip
set ip precedence 5
class control
set ip precedence 5
class class-default
set ip precedence 0
!
interface Ethernet0/0
service-policy input mqc
```

In dit configuratievoorbeeld wordt elk verkeer dat overeenkomt met Toegangscontrolelijst (ACL) 100 geclassificeerd als "class voip" en ingesteld met IP voorrang 5. Dit betekent dat de drie meest significante bits van de IP ToS-byte op 101 zijn ingesteld. ACL 100 komt overeen met de gebruikelijke UDP-poorten die door VoIP worden gebruikt. Op dezelfde manier komt ACL 101 overeen met H.323 signaleringsverkeer (Transmission Control Protocol (TCP) poort 1720). Al het andere verkeer wordt ingesteld met IP voorrang 0. Het beleid wordt "mqc" genoemd. Het wordt toegepast op inkomend verkeer op Ethernet 0/0.

prioritering

Nadat elke hop in het netwerk de VoIP-pakketten kan classificeren en identificeren (door middel van poort/adresinformatie of door de ToS-byte), voorzien deze hop dan in elk VoIP-pakket van de

vereiste QoS. Op dat punt, vorm speciale technieken om voorrang te verlenen in het wachtrij om ervoor te zorgen dat grote gegevenspakketten de transmissie van spraakgegevens niet belemmeren. Dit is meestal vereist voor langzamere WAN-links waar een grote kans op congestie is. Zodra al het interessante verkeer in QoS-classes is geplaatst op basis van hun QoS-vereisten, bieden u bandbreedtegaranties en prioriteitservice via een intelligent output Wachtmechanisme. Er is een prioriteitswachtrij voor VoIP vereist.

OPMERKING: Gebruik een wachtmechanisme dat VoIP effectief een hoge prioriteit geeft. Low Latency Queuing (LLQ) wordt echter aanbevolen, omdat het flexibel en gemakkelijk te configureren is.

LLQ gebruikt de modulaire QoS CLI-configuratiemethode om prioriteit aan bepaalde klassen te geven en gegarandeerde minimumbandbreedte voor andere klassen te bieden. Tijdens perioden van congestie wordt de prioriteitswachtrij in de geconfigureerde snelheid ingeschakeld, zodat het prioriteitsverkeer niet alle beschikbare bandbreedte opgebruikt. (Als het prioriteitsverkeer de bandbreedte monopoliseert, voorkomt het bandbreedte-garanties voor andere klassen om te worden bereikt.) Als u LLQ correct verstrekt, zou het verkeer dat in de prioriteitenrij gaat nooit het gevormde tarief moeten overschrijden.

LLQ staat ook toe dat de diepten van de rij worden gespecificeerd om te bepalen wanneer de router pakketten moet laten vallen als er te veel pakketten zijn die in een bepaalde rij van klassen wachten. Er is ook een **class-default** opdracht die wordt gebruikt om de behandeling te bepalen van al het verkeer dat niet door een geconfigureerde klasse is ingedeeld. De class-default wordt ingesteld met een **flankerende** opdracht. Dit betekent dat elke niet-gerubriceerde stroom een ongeveer gelijk aandeel van de resterende bandbreedte krijgt.

Dit voorbeeld toont hoe te om LLQ te configureren. Raadpleeg voor meer informatie de [Low Latency Queuing](#):

```
access-list 100 permit udp any any range 16384 32000
access-list 101 permit tcp any any eq 1720
access-list 102 permit tcp any any eq 80
access-list 103 permit tcp any any eq 23
!
class-map match-all voip
match access-group 100
class-map match-all voip-control
match access-group 101
class-map match-all data1
match access-group 102
class-map match-all data2
match access-group 103
!
policy-map llq
class voip
priority 32
class voip-control
bandwidth 8
class data1
bandwidth 64
class data2
bandwidth 32
class class-default
fair-queue
!
interface Serial1/0
bandwidth 256
```

In dit voorbeeld wordt elk verkeer dat ACL 100 aanpast geclassificeerd als "class voip" (dat spraakverkeer). Het krijgt een hoge prioriteit tot 32 kbps. ACL 100 (ACL) komt overeen met de gebruikelijke UDP-poorten die door VoIP worden gebruikt. Toegangslijst 101 past het H.323 signaleringsverkeer aan (TCP poort 1720). De klasse data1 past web verkeer aan (TCP poort 80 zoals gezien in Toegangslijst 102) en garandeert 64 kbps. Class Data2 komt overeen met Telnet-verkeer (TCP-poort 23 zoals gezien in ACL 103) en garandeert 32 kbps. De standaardklasse wordt ingesteld om een gelijk deel van de resterende bandbreedte aan niet-gerubriceerde stromen te geven. Het beleid wordt "llq" genoemd. Het wordt toegepast op uitgaande verkeer op Serial1/0, dat een totale bandbreedte van 256 kbps heeft.

Opmerking: Standaard moet de totale gegarandeerde bandbreedte en prioriteitsbandbreedte voor alle klassen kleiner zijn dan 75% van de interfacebandbreedte. Wijzig dit percentage door de **max-gereserveerde opdracht voor bandbreedte uit te geven**.

In deze tabel worden verschillende mechanismen voor softwarewachtrij met hun respectievelijke voordelen en beperkingen vergeleken.

Software Queuing-mechanismen	Beschrijving	Voordelen	Beperkingen
First-In-First-Out (FIFO)	Pakketten arriveren en verlaten de rij in precies dezelfde volgorde.	Eenvoudige configuratie en snelle bediening.	Geen prioriteitsservice- of bandbreedte garanties mogelijk. ¹
Weighted Fair Queuing (WFQ)	Een hashing-algoritme die in afzonderlijke rijen stroomt waar gewichten worden gebruikt om te bepalen hoeveel pakketten tegelijkertijd worden onderhouden. U definieert gewichten door IP-voorrang en DSCP-waarden in	Eenvoudige configuratie. Standaard op koppelingen kleiner dan 2 Mbps.	Geen prioriteitsservice- of bandbreedte garanties mogelijk. ¹

	te stellen.		
Aangepaste wachtrij (CQ)	<p>Het verkeer wordt geclassificeerd in meerdere wachtrijen met configureerbare wachtrijen. De wachtrijlimieten worden berekend op basis van gemiddelde pakketgrootte, Max. Transmissie Eenheid (MTU) en het percentage van de bandbreedte dat moet worden toegewezen. De wachtrijen worden (in aantal bytes) voor elke wachtrij vrijgegeven. Daarom levert het de toegewezen bandbreedte statistisch.</p>	<p>Is al een paar jaar beschikbaar. Het maakt een benadering van bandbreedte-toewijzing voor verschillende wachtrijen mogelijk.</p>	<p>Onderhoud met voorrang is niet mogelijk. De garanties van de bandbreedte zijn bij benadering. Er zijn een beperkt aantal rijen. Configuratie is relatief moeilijk.¹</p>
Prioriteitswachtrij (PQ)	<p>Het verkeer wordt geclassificeerd in wachtrijen met hoge, gemiddelde, normale en lage prioriteit. Het</p>	<p>Is al een paar jaar beschikbaar. Het biedt prioritaire diensten.</p>	<p>Hogere prioriteit van het verkeer hongert de lagere prioriteitsrijen van bandbreedte. Er zijn geen bandbreedte</p>

	prioriteitsverkeer wordt eerst onderhouden, gevolgd door middelgrote, normale en lage prioriteit.		garanties mogelijk. ²
Op klasse gebaseerde Weighted Fair Queuing (CBWFQ)	De modulaire QoS CLI wordt gebruikt om het verkeer te classificeren. Geclassificeerd verkeer wordt in gereserveerde bandbreedte wachtrijen of een standaard ongereserveerde wachtrij geplaatst. Een plannerservices de wachtrijen op basis van gewichten zodat de bandbreedte garanties worden nageleefd.	Gelijkaardig aan LLQ, behalve dat er geen prioriteit wachtrij is. Eenvoudige configuratie en mogelijkheid om bandbreedtegaranties te bieden.	Onderhoud met voorrang is niet mogelijk. ³
Prioritaire wachtrij voor Weighted Fair Queuing (PQ-WFQ), ook wel IP	Een enkele interfaceopdracht wordt gebruikt om voorrang te verlenen bij het onderhouden	Eenvoudig, één commandoconfiguratie. Hier vindt u prioritair onderhoud aan RTP-pakketten.	Al het andere verkeer wordt behandeld met WFQ. Real-Time Conferencing

<p>RTP-prioriteit genoemd</p>	<p>n van diensten aan alle UDP-pakketten die bestemd zijn om zelfs poortnummers binnen een bepaald bereik in te voeren.</p>		<p>g Protocol (RTCP)-verkeer wordt niet geprioriteerd . Geen gegarandeerde bandbreedte-capaciteit.⁴</p>
<p>LLQ, eerder genoemd op klasse-gebaseerde Weighted Fair Queuing (PQCBWFQ)</p>	<p>De modulaire QoS CLI met prioriteitswachtrij wordt gebruikt om verkeer te classificeren . Geclassificeerd verkeer wordt in een prioriteitsrij, gereserveerde bandbreedtewachtrijen of een standaard ongereserveerde rij geplaatst. Een planner-services de wachtrijen op basis van gewichten zodat het prioriteitsverkeer eerst wordt verzonden (tot een bepaalde beveiligde limiet tijdens congestie) en de</p>	<p>Eenvoudige configuratie. Mogelijkheid om prioriteit aan meerdere klassen van verkeer te verlenen en bovengrenzen op prioritair bandbreedtegebruik te geven. U kunt ook bandbreedte-gegarandeerde klassen en een standaardklasse configureren.</p>	<p>Geen mechanisme om meerdere prioriteitsniveaus te bieden, al het prioriteitsverkeer wordt door dezelfde prioriteitswachtrij gestuurd. Afzonderlijke prioriteitsklassen kunnen afzonderlijke bandbreedte-grenzen met hoge prioriteit hebben tijdens congestie. Het delen van een prioriteitswachtrij tussen toepassingen kan echter jitter introduceren.⁴</p>

	bandbreedte garanties worden gehaald.		
--	------------------------------------------------	--	--

1. Niet geschikt voor stem.
2. Moet bandbreedte gegarandeerd zijn voor spraak.
3. Er moet op worden gelet dat de nodige vertraging wordt opgelopen.
4. Genoeg voor stem.

Servische vertraging

Zelfs als het in de wachtrij plaatsen op zijn best werkt en prioriteit geeft aan spraakverkeer, zijn er tijden wanneer de prioriteitswachtrij leeg is en er service wordt verleend aan een pakketreis van een andere klasse. Packets van gegarandeerde bandbreedte-klassen moeten worden onderhouden op basis van hun geconfigureerde gewicht. Als een prioritair stempakket in de uitvoerrij aankomt terwijl deze pakketten worden onderhouden, kan het spraakpakket een aanzienlijke hoeveelheid tijd wachten voordat het wordt verzonden. Spraakpakketten ervaren serialbreer vertraging wanneer ze achter grotere gegevenspakketten moeten wachten.

De vertraging van de servering kan de ergste vorm van jitter voor spraakpakketten introduceren. Als de spraakpakketten achter een gegevenspakket van 1500 bytes moeten wachten, op een langzamere link, vertaalt dit zich naar een grote vertraging. De serialbreer vertraging is aanzienlijk anders als het gegevenspakket 80 bytes is, zoals in dit voorbeeld wordt getoond:

- Serialbreer vertraging op een link van 64 kbps door een pakket van 1500 bytes = $1500 \cdot 8 / 64000 = 187,5$ ms.
- Serialbreer vertraging op een link van 64 kbps door een pakket van 80 bytes = $80 \cdot 8 / 64000 = 10$ ms.

Daarom moet een spraakpakket potentieel tot 187,5 ms wachten voordat het wordt verzonden als het achter een enkel pakket van 1500 bytes op een link van 64 kbps klem komt te zitten. Aan de andere kant, moet een ander stempakket slechts 10 ms op de doelgateway wachten. Dit levert een groot probleem op vanwege de variantie in de interpakketvertraging. Op de gateway van oorsprong, worden de spraakpakketten gewoonlijk om de 20 ms verzonden. Met een budget van 150 ms voor uitstel van eind tot eind is een gat van meer dan 180 ms onaanvaardbaar.

Invoering van een fragmentatiemechanisme dat ervoor zorgt dat de grootte van één transmissie-eenheid minder dan 10 ms bedraagt. Alle pakketten met meer dan 10 ms vertraging moeten worden gefragmenteerd in 10 ms stukjes. Een stuk of fragment van 10 ms is het aantal bytes dat over de link in 10 ms wordt verzonden. Bereken de grootte met behulp van de verbindingssnelheid, zoals in dit voorbeeld wordt getoond:

- Fragmentation size = $(0,01 \text{ seconden} \cdot 64.000 \text{ bps}) / (8 \text{ bits/bytes}) = 80$ bytes

Het duurt 10 ms om een pakket met 80 bytes te verzenden of een fragment over een link van 64 kbps.

In het geval van meerdere ATM of Frame Relay Permanent Virtual Circuits (PVC's) op één fysieke interface, moet u de fragmentatiewaarden (op alle PVC's) configureren op basis van het PVC dat de laagste beschikbare bandbreedte heeft. Als er bijvoorbeeld drie PVC's zijn met een gegarandeerde bandbreedte van 512 kbps, 128 kbps en 256 kbps, moet u alle drie PVC's configureren met een fragment van 160 bytes (de laagste snelheid is 128 kbps en vereist een 160-byte-fragment). Deze waarden worden aanbevolen voor verschillende link snelheden:

Link Speed (kbps)	Fragmentation Size (bytes)
56	70
64	80
128	160
256	320
512	640
768	960
1024	1280
1536	1920

Toelichting: Een fragmentatie is niet vereist als het fragment groter is dan de grootte van de link MTU. Bijvoorbeeld, voor een T1 verbinding met een MTU van 1500 bytes is de grootte van het fragment 1920 bytes. Daarom is er geen fragmentatie nodig. De grootte van de pakketfragmentatie mag nooit lager zijn dan de VoIP-pakketgrootte. VoIP-pakketten niet splitsen. Het fragmenteren van deze pakketten veroorzaakt talrijke vraag opstelling en kwaliteitskwesities.

Momenteel zijn er drie mechanismen voor verbrokkeling en vervlechting van verbindingen beschikbaar. Voor verdere uitleg van verschillende vertragingen die in een pakketnetwerk zijn geïntroduceerd, raadpleeg [Vertraging in Packet Voice Networks](#). In deze tabel worden de voordelen en beperkingen genoemd:

Mechanisme voor Link Fragmentation and Interleaving (LFI)	Beschrijving	Voordelen	Beperkingen
MTU-fragmentatie met WFQ	Opdracht Interfaceniveau om MTU of IP MTU grootte te veranderen. Gebruikt om grote IP-pakketten te fragmenteren naar een bepaalde grootte van MTU. LFI gebruikt WFQ om realtime pakketten tussen de fragmenten te onderbreken.	Eenvoudige configuratie.	De fragmenten worden alleen door de ontvangende toepassing opnieuw samengevoegd. Daarom wordt het netwerk niet efficiënt gebruikt. Alleen IP-pakketten met het Do not Fragment (DF)-bit kunnen fragmentati

			e niet goed verwerken. Zeer processorintensief. Niet aanbevolen.
Multilink Point-to-Point Protocol (MLPPP) -LFI	Op point-to-point seriële links moet MLPPP eerst worden geconfigureerd en vervolgens moet een fragmentatiegrootte in ms worden ingesteld. Interleaving moet ook zijn ingeschakeld op de multilink-interface.	Pakketten zijn gefragmenteerd aan het ene uiteinde van de verbinding en aan het andere uitgezet. Verschillende links kunnen worden gecombineerd om als een grote virtuele buis te fungeren.	Alleen beschikbaar op koppelingen die voor PPP zijn geconfigureerd. Oplossingen voor PPP via Frame Relay of PPP via ATM worden ook ondersteund in Cisco IOS-software release 12.1(5)T of hoger.
Frame Relay Fragmentation (FRF.12)	Op Frame Relay PVC's moet de opdracht frame-relais traffic-shaping ingeschakeld zijn en moet een fragmentatiegrootte ingesteld zijn onder de map-klasse.	Pakketten zijn gefragmenteerd aan de ene kant van het PVC en aan de andere kant geherassembleerd.	Alleen beschikbaar op Frame Relay PVC's met de opdracht frame-relais die is ingeschakeld.

VAD

Een regelmatig spraakgesprek bestaat uit verschillende momenten van stilte. Een typische spraakgesprek bestaat uit 40 tot 50 procent stilte. Aangezien er geen stem door het netwerk gaat voor 40% van een stemvraag, kan enige bandbreedte worden gered door VAD in te zetten. Met VAD zoekt de poort naar openingen in de spraak. Deze ruimtes worden vervangen door comfort ruis (achtergrondruis). Hierdoor wordt een hoeveelheid bandbreedte opgeslagen. Er is echter sprake van een ruil. Er is een kleine tijd (in de volgorde van milliseconden), voordat de codecs spraakactiviteit detecteren, gevolgd door een periode van stilte. Deze kleine tijd resulteert in het voorste knippen van ontvangen stem. Om activering tijdens zeer korte pauzes te voorkomen en om voor het knippen te compenseren, wacht VAD ongeveer 200 ms na toespraak stop voordat de

overdracht stopt. Bij het opnieuw opstarten van de transmissie, zitten de voorgaande 5 spraakmodules in combinatie met de huidige toespraak. VAD schakelt zichzelf automatisch uit op een oproep als omgevingslawaai voorkomt dat het onderscheid maakt tussen spraak- en achtergrondlawaai. Als de bandbreedte echter geen probleem is, schakelt u de VAD uit.

Tune VAD-parameters

Er zijn twee parameters die het functioneren van de VAD dicteren. Dit zijn de **muziek-drempelwaarden** en **spraak-tijd** opdrachten.

muziekdrempel

Er wordt een initiële drempel vastgesteld voor welke gevallen waarin VAD actief moet worden. Dit wordt gecontroleerd door het bepalen van de **muziek-drempelwaarde** *drempelwaarde_value* opdracht op een spraak-poort, zoals getoond in dit voorbeeld. Het bereik hiervoor is van -70 Decibel per Milliwatt (dBm) tot -30 dBm. De standaardwaarde voor dit is -38 dBm. Het configureren van een lagere waarde (naar -70 dBm) resulteert in het actief worden van VAD bij een veel lagere signaalsterkte (het volume moet echt laag dalen voordat het als stilte wordt beschouwd). Het configureren van een hogere waarde (dichterbij -30 dBm) resulteert in VAD actief worden voor zelfs een kleine daling in de sterkte van het spraaksignaal. Het drijft de uitspeelbuis om comfortruis vaak te spelen. Dit leidt echter soms tot een kleine knippering van het geluid.

```
3640-6#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
3640-6(config)#voice-port 3/0/0
3640-6(config-voiceport)#music-threshold ?
WORD Enter a number b/w (-70 to -30)
3640-6(config-voiceport)#music-threshold -50
3640-6(config-voiceport)#end
3640-6#
3640-6#show run | be voice-portvoice-port 3/0/0 music-threshold -50
```

spreektijd

Zodra de VAD actief wordt, wordt de component van achtergrondlawaai en comfort lawaai gecontroleerd door de **opdracht** *timer_timer_waarde* te configureren onder de mondiale configuratie, zoals in dit voorbeeld wordt getoond. Dit is de vertragingstijd in milliseconden voor stilzwijgdetectie en suppressie van spraakpakkettransmissie. De standaardwaarde voor de vertragingstijd is 250 msec. Dit betekent dat binnen 250 msec, comfort ruis begint. Het bereik van deze timer is 250 msec tot 6536 msec. Als hiervoor een hoge waarde is ingesteld, komt comfortgeluid veel later in werking (achtergrondruis blijft spelen). Als dit is ingesteld voor 6536 msec, dan wordt het comfortgeluid uitgeschakeld. Een hogere waarde voor deze timer is gewenst voor een soepele overgang tussen achtergrondruis en comfortruis. Het nadelig om de **spraak vad-time** opdracht op een hoog niveau te configureren is dat de gewenste 30 tot 35 procent bandbreedte-besparing niet wordt bereikt.

```
3640-6#
3640-6#
3640-6#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
3640-6(config)#voice vad-time ?
<250-65536> milliseconds
```

```
3640-6(config)#voice vad-time 750
3640-6(config)#end
3640-6#
3640-6#
3640-6#
3640-6#show run | be vad-time voice vad-time 750
```

Typische Configuratievoorbeelden voor QoS

Een typisch scenario voor het opzetten van VoIP vraag is over een frame-relais verbinding of over een PPP verbinding. Dit zijn configuratievoorbeelden voor deze scenario's.

VoIPoFR - QoS-configuratievoorbeeld

In dit voorbeeld (dat alleen relevante delen van de configuratie bevat), wordt aangenomen dat de snelheid van de frame-relais 256 kbps is. De gegarandeerde Committed Information Rate (CIR) voor PVC 100 is 64 kbps en voor PVC 200 is 192 kbps. PVC 100 wordt gebruikt om zowel gegevens als spraak te dragen. PVC 200 wordt uitsluitend gebruikt om gegevens over te brengen. Er zijn op elk moment maximaal vier gelijktijdige spraakoproepen. Configuratie fragmentatie op beide PVC's gebaseerd op de CIR van de laagst-bandbreedte-stem-PVC (PVC-dragende stem). Gebaseerd op de voorbeelden in dit document betekent dit dat de fragmentatiegrootte wordt bepaald op basis van de CIR van PVC 100 (dat 64 kbps is). Zoals in de tabel van de sectie "Vertraging van de serieel" wordt aangegeven, is voor een koppeling van 64 kbps een fragmentatiegrootte van 80 bytes vereist. Dezelfde fragmentatiegrootte moet voor PVC 200 worden geconfigureerd.

Voor meer informatie over de configuratie van VoIP via Frame Relay, raadpleeg [VoIP via Frame Relay met Quality of Service \(Fragmentation, Traffic Shaping, LLQ/IP RTP-prioriteit\)](#).

```
3660-1#show run
Building configuration...
!
class-map match-any voip
match ip rtp 16384 16383
match ip dscp 26 46
class-map match-all voip-control
match access-group 101
!
!
policy-map VoIPoFR
class voip
priority 48
class voip-control
bandwidth 8
class class-default
fair-queue
!
voice call send-alert
voice rtp send-recv
!
!
interface Serial4/0:0
bandwidth 256
no ip address
encapsulation frame-relay
frame-relay traffic-shaping
```

```

!
interface Serial4/0:0.1 point-to-point
bandwidth 64
ip address 10.10.10.10 255.255.255.0
frame-relay ip rtp header-compression
frame-relay interface-dlci 100
  class voice
!
interface Serial4/0:0.2 point-to-point
bandwidth 192
ip address 20.20.20.20 255.255.255.0
frame-relay interface-dlci 200
class data
!
map-class frame-relay data
frame-relay fragment 80
frame-relay adaptive-shaping becn
frame-relay cir 256000
frame-relay bc 32000
frame-relay be 0
frame-relay mincir 192000
frame-relay fair-queue
!
map-class frame-relay voice
frame-relay fragment 80
no frame-relay adaptive-shaping
frame-relay cir 64000
frame-relay bc 640
frame-relay be 0
frame-relay mincir 64000
service-policy output VoIPoFR
!
!
access-list 101 permit tcp any any eq 1720
!
!
voice-port 3/1/0
!
voice-port 3/1/1
!
!
dial-peer voice 10 voip
incoming called-number .
destination-pattern 1408.....
session target ipv4:10.10.10.11
dtmf-relay h245-signal h245-alphanumeric
no vad
!
dial-peer voice 20 pots
destination-pattern 1234
port 3/1/0
!
dial-peer voice 21 pots
destination-pattern 5678
port 3/1/1

```

VoIP over PPP - QoS-configuratievoorbeeld

In dit voorbeeld (dat alleen relevante onderdelen van de configuratie bevat) wordt aangenomen dat de QoS moet worden geconfigureerd voor een point-to-point fractionele T1 controller (dat twaalf kanalen heeft). Er zijn op elk moment maximaal vier gelijktijdige spraakoproepen. De configuratietaak omvat het configureren van deze seriële interface met PPP-insluiting, het maken

van een onderdeel van een multilink-groep, het maken van een multilink-interface (die tot dezelfde multilink-groep behoort) en het configureren van alle QoS op de multilink-interface. Voor meer informatie over de configuratie van VoIP via PPP, raadpleeg [VoIP over PPP Links met Quality of Service \(LLQ/IP RTP Priority, LFI, cRTP\)](#).

```
3660-1#show run
Building configuration...
!
class-map match-any voip
match ip rtp 16384 16383
match ip dscp 26 46
class-map match-all voip-control
match access-group 101
!
!
policy-map VoIPoPPP
class voip
priority 48
class voip-control
bandwidth 8
class class-default
fair-queue
!
voice call send-alert
voice rtp send-recv
!
!
interface Multilink7
bandwidth 768
ip address 10.10.10.10 255.255.255.0
ip tcp header-compression iphc-format
service-policy output VoIPoPPP
no cdp enable
ppp multilink
ppp multilink fragment-delay 10
ppp multilink interleave
multilink-group 7
ip rtp header-compression iphc-format
!
!
interface Serial4/0:0
bandwidth 768
no ip address
encapsulation ppp
no fair-queue
ppp multilink
multilink-group 7
!
!
access-list 101 permit tcp any any eq 1720
!
voice-port 3/1/0
!
voice-port 3/1/1
!
!
dial-peer voice 10 voip
incoming called-number .
destination-pattern 1408.....
session target ipv4:10.10.10.11
dtmf-relay h245-signal h245-alphanumeric
```

```

no vad
!
dial-peer voice 20 pots
destination-pattern 1234
port 3/1/0
!
dial-peer voice 21 pots
destination-pattern 5678
port 3/1/1
!

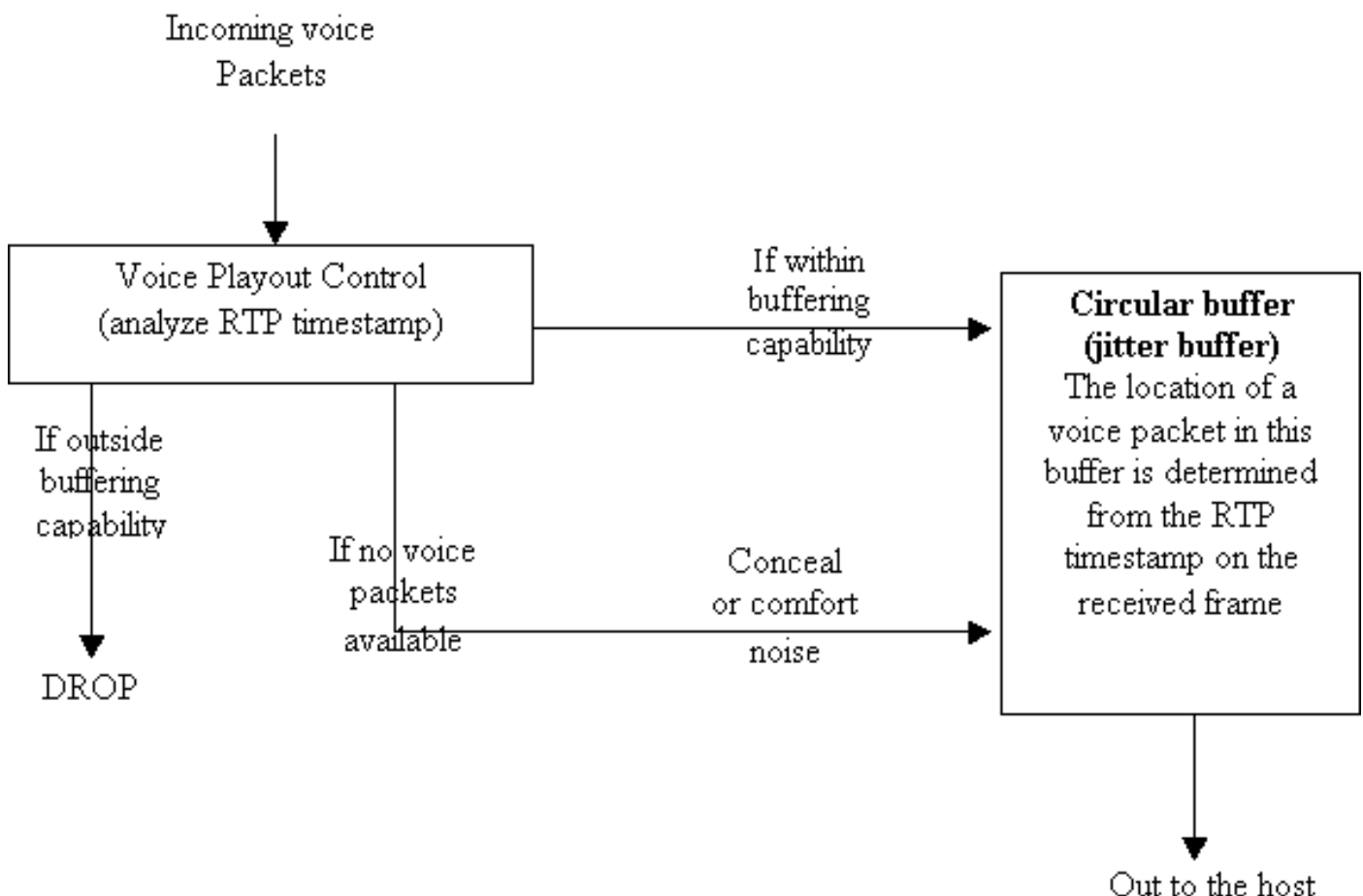
```

Mechanisme voor jitter en playout

Er zijn altijd enkele ongecontroleerde entiteiten in een netwerk die bijdragen aan verdere vertragingen en jitter in de ontvangen spraakpakketten. Door de jitterbuffer op de eindgateway te wijzigen wordt de ongecontroleerde jitter opgelost in het spraaknetwerk.

Indelingsmechanisme

De jitterbuffer is een tijdbuffer. De terminating poort biedt de mogelijkheid het uitloopmechanisme doeltreffender te maken. Dit is een functioneel schema van het uitloopmechanisme:



Wanneer de playout controle een stempakket ontvangt, analyseert het de tijdstempel RTP. Als het stempakket uitgesteld is voorbij de capaciteit van de jitter buffer, dan wordt het pakje meteen ingetrokken. Als het pakje binnen de bufferfunctie valt, wordt het in de jitterbuffer geplaatst. De locatie van dit pakje in de jitter-buffer is afhankelijk van de RTP-tijdstempel die voor dat pakje is berekend. Als er geen beschikbaar spraakpakket is, probeert de uitrolcontrole het te verbergen (voorspeld het uitbetaalde pakje). Als VAD is ingeschakeld, wordt comfort ruis weergegeven.

De verantwoordelijkheid van de playout controle is om de gebeurtenissen van verloren pakketten, gedupliceerde pakketten, gecorrumpeerde pakketten, en uit-van-reeks pakketten aan te pakken. Deze gebeurtenissen worden afgehandeld door de tijd uit te lijnen van de geïttereerde spraakpakketten, comfortruis (als VAD is geconfigureerd) of zelfs dubbele toonfrequentie (DTMF) tonen die aan de host moeten worden afgespeeld.

Het verbergen van een spraakpakket gebeurt door voorspelling of door stilzwijgende verzwijging. Verhulling van voorspelling is gebaseerd op het vorige pakket en het volgende pakket (indien beschikbaar). Het werkt het beste met lage bitratecodecs (5 kbps tot 16 kbps). Verlies van spraakpakketten voor een hoge bitratecodec (32 kbps tot 64 kbps) kan mogelijk resulteren in een slechte verzwijging van de voorspellingen. Voorspellingsverzwijging begint bij lage en af en toe vertragingen of een kleiner aantal pakketverlies. Te veel voorspellingen verzwijgen kan leiden tot robotspraakqualiteit. Stille verhullen is de ergste vorm van voorspelling-verzwijging. Het speelt in als er geen informatie beschikbaar is om te voorspellen. Het is gewoon een achterdocht. Het begint wanneer er grote vertragingen zijn en meer pakketverlies. Te veel stille verzwijging leidt tot een hachelijke spraakqualiteit. Voorspeld verzwijging is goed voor 30 minuten, waarna het stilzwijgen in werking treedt.

Jitter Buffer

De jitterbuffer wordt opgesloten door een hoog watermerk en een laag watermerk. Het hoge watermerk is de hoogste tijdslijmiet waarbinnen een pakje naar verwachting voor een on-time playout zal aankomen. Pakketten die na het hoge watermerk arriveren worden gemarkeerd als late pakketten of verloren pakketten. Het lage watermerk is de minimumtijd waarbinnen een pakje voor een tijdelijke uitloop wordt verwacht. Pakketten die arriveren vóór het lage watermerk worden als vroege pakketten beschouwd (de pakketten kunnen nog op tijd worden uitgespeeld).

Als de terminerende gateway een toename van de aankomst van de late pakketten blijft zien, wordt het hoge watermerk verhoogd. Deze waarde voor een hoog watermerk blijft tijdens de duur van de oproep gelijk. Dit wordt verhoogd tot een maximum dat in de configuratie wordt gedefinieerd. Op een zelfde manier, de eindgateway observeert het aantal vroege ontvangen pakketten. Als deze pakketten de gateway veelvuldig beginnen te vullen, vermindert deze het lage watermerk. Deze waarde blijft tijdens de duur van de oproep gelijk. Deze modus van de jitterbuffer wordt "adaptieve modus" genoemd, waarbij de eindgateway haar jitterbuffer aanpast op basis van het verkeerspatroon. De andere modus is "vaste modus". In de vaste modus is er één beginwaarde voor het waterniveau en het watermerk. Deze waarde is gebaseerd op de geschatte ontvangen vertraging (zie het gedeelte [Spraakoproepen <poort-nummer>](#) van dit document).

Voor verdere details op jitter buffer, verwijst naar [Understanding Jitter in Packet Voice Networks \(Cisco IOS-platforms\)](#).

Timer en jitter identificeren

In dit gedeelte wordt beschreven hoe u de tekst in uw netwerk kunt identificeren.

roep actieve stem tonen

De **show call actieve stem korte** opdracht geeft veel informatie over een lopend gesprek. Deze uitvoer toont een aantal belangrijke punten die van deze opdracht worden geleerd:

```
11E4 : 2170927hs.1 +600 pid:10 Answer 1000 active
dur 00:08:43 tx:26157/522967 rx:7044/139565
Tele 3/0/0:9: tx:151310/755/0ms g729r8 noise:-62 acom:0 i/0:-56/-48 dBm
11E4 : 2171198hs.1 +329 pid:20 Originate 2000 active
dur 00:08:43 tx:7044/139565 rx:26165/523127
IP 30.30.30.29:18682 rtt:51ms pl:148590/290ms lost:0/0/15 delay:65/60/132ms g729r8
```

Van de **show roep actieve stem korte** opdrachtoutput, zie je dat wat dan ook ontvangen wordt op het Telephony been (rx:7044) wordt verzonden naar het IP-been (tx:7044). Hetzelfde geldt voor pakketten die op de IP-poten (26165) worden ontvangen en die naar het Telephony-been (26157) worden verzonden. Het verschil in het aantal pakketten die op het IP been worden ontvangen versus het aantal pakketten die op het been van de Telefonie worden verzonden wordt bijgedragen aan late pakketten die het niet in tijd maken.

Deze output van de **show roept actief stembevel** (zonder het "korte"sleutelwoord), wijst op verdere details over parameters die direct jitter identificeren.

```
GapFillWithSilence=850 ms
GapFillWithPrediction=9230 ms
GapFillWithInterpolation=0 ms
GapFillWithRedundancy=0 ms
```

[Bel <port-number>](#)

De opdracht **Show Voice Call Port-number** biedt nuttige informatie. Zorg ervoor dat u in de gateway bent ingesloten of dat u via Telneted in een gateway bent, zorg er dan voor dat u de opdracht voor de **terminalmonitor** vanaf het exec-niveau hebt afgegeven.

Opmerking: deze opdracht is niet beschikbaar op de AS5x00/AS5x50-platforms.

In deze output is de waarde voor Rx Delay East (ms) 71. Dit is de huidige jitter buffer waarde. Hierop wordt een waarde voor het hoge en lage watermerk in mindering gebracht. Een gemiddelde beginwaarde voor het hoge waterniveau is 70 msec, terwijl die voor het lage waterniveau 60 msec bedraagt. Zodra een aanvankelijke waarde wordt ingesteld houdt de gateway sporen van om het even welke vroege pakketten of late ontvangen pakketten bij. Zoals je hier kunt zien in de output zie je dat het verzwijgen van de voorspelling bijna 250 ms bedraagt, terwijl het verzwijgen van de stilte 30 ms hoog is. Er is altijd een hogere waarde voor het verbergen van voorspellingen, omdat het verzwijgen van het zwijgen slechts een erger geval van verhulling van het voorspellingsscenario is. Bij elke voorspelling-daling is er een toename in de teruggooi van de bufferoverloop.

Als u de buffer ziet weggooien, betekent dit niet noodzakelijk dat u een verhoging van het hoge watermerk ziet. Het hoge watermerk is de bovengrens van de jitterbuffer. Het verandert alleen als er een trend wordt waargenomen. Met andere woorden: er moet een onafgebroken stroom van late pakketten zijn. Dit leidt tot een verhoging van de jitterbuffer. In de productie hier is een dergelijke trend zichtbaar. Daarom wordt het hoge waterniveau verhoogd van 70 msec tot 161 msec. Als deze waarde niet wordt gewijzigd (en als u nog steeds 14 late pakketten ziet), impliceert dit dat dit sporadisch late pakketten zijn en geen trend vormt.

Vanaf de output van de **show vraag actieve stem** opdracht, kijk uit voor verloren pakketten. Voor elk verloren pakje, zie je twee pakketten die niet in volgorde zijn. Dit wordt gezien op de RX Non-Seq Pkts output. Aangezien het geen positieve waarde is, wordt geconcludeerd dat er ook geen sprake is van pakketverlies.

```

3640-6# ***DSP VOICE TX STATISTICS***
Tx Vox/Fax Pkts: 195, Tx Sig Pkts: 0, Tx Comfort Pkts: 10
Tx Dur(ms): 192070, Tx Vox Dur(ms): 388, Tx Fax Dur(ms): 0
***DSP VOICE RX STATISTICS***
Rx Vox/Fax Pkts: 9604, Rx Signal Pkts: 0, Rx Comfort Pkts: 0
Rx Dur(ms): 192070, Rx Vox Dur(ms): 191560, Rx Fax Dur(ms): 0
Rx Non-seq Pkts: 0, Rx Bad Hdr Pkts: 0
Rx Early Pkts: 0, Rx Late Pkts: 14
***DSP VOICE VP_DELAY STATISTICS***
Clk Offset(ms): 0, Rx Delay Est(ms): 71
Rx Delay Lo Water Mark(ms): 60, Rx Delay Hi Water Mark(ms): 161
***DSP VOICE VP_ERROR STATISTICS***
Predict Conceal(ms): 250, Interpolate Conceal(ms): 0
Silence Conceal(ms): 30, Retroact Mem Update(ms): 0
Buf Overflow Discard(ms): 500, Talkspurt Endpoint Detect Err: 0
***DSP LEVELS***
TDM Bus Levels(dBm0): Rx -49.9 from PBX/Phone, Tx -41.7 to PBX/Phone
TDM ACOM Levels(dBm0): +2.0, TDM ERL Level(dBm0): +11.1
TDM Bgd Levels(dBm0): -58.9, with activity being voice
***DSP VOICE ERROR STATISTICS***
Rx Pkt Drops(Invalid Header): 0, Tx Pkt Drops(HPI SAM Overflow): 0

```

Neem de TX-computten en RX-computten in acht. Vanaf de voorbeeldoutput wordt geconcludeerd dat de telefoon die op deze router wordt aangesloten meestal stil houdt aangezien u veel TX Comfort Pkts hebt. Tegelijkertijd heb je nul Rx Comfort Pkts, wat betekent dat het andere uiteinde voortdurend spreekt.

Vergelijk de uitvoer hier met de vorige opdrachtoutput. Er is een verhoogd aantal RX-late Pkts (van 14 tot 26). Er is echter geen verhoging van de hoge waarde van de watermerken. Dit geeft aan dat de 12 pakketten sporadisch worden vertraagd. De teruggooi van de buffer wordt verhoogd tot 910 msec. Aangezien er echter geen trend wordt waargenomen, wordt het hoge waterpeil niet verhoogd.

In de output hier, hebt u een x Early Pkts: 3. Dit betekent dat een pakje veel aankomt voordat het wordt verwacht. Zoals te zien is vanuit de output hier, heeft de Jitter buffer zichzelf uitgestrekt om op meer vroege pakketten plaats te maken door het lage watermerk van 60 tot 51 te verlagen.

```

3640-6# ***DSP VOICE TX STATISTICS***
Tx Vox/Fax Pkts: 209, Tx Sig Pkts: 0, Tx Comfort Pkts: 11
Tx Dur(ms): 337420, Tx Vox Dur(ms): 416, Tx Fax Dur(ms): 0
***DSP VOICE RX STATISTICS***
Rx Vox/Fax Pkts: 16843, Rx Signal Pkts: 0, Rx Comfort Pkts: 1
Rx Dur(ms): 337420, Rx Vox Dur(ms): 335920, Rx Fax Dur(ms): 0
Rx Non-seq Pkts: 0, Rx Bad Hdr Pkts: 0
Rx Early Pkts: 3, Rx Late Pkts: 26
***DSP VOICE VP_DELAY STATISTICS***
Clk Offset(ms): 0, Rx Delay Est(ms): 72
Rx Delay Lo Water Mark(ms): 51, Rx Delay Hi Water Mark(ms): 161
***DSP VOICE VP_ERROR STATISTICS***
Predict Conceal(ms): 510, Interpolate Conceal(ms): 0
Silence Conceal(ms): 70, Retroact Mem Update(ms): 0
Buf Overflow Discard(ms): 910, Talkspurt Endpoint Detect Err: 0
***DSP LEVELS***
TDM Bus Levels(dBm0): Rx -51.5 from PBX/Phone, Tx -44.1 to PBX/Phone
TDM ACOM Levels(dBm0): +2.0, TDM ERL Level(dBm0): +11.9
TDM Bgd Levels(dBm0): -61.3, with activity being voice
***DSP VOICE ERROR STATISTICS***
Rx Pkt Drops(Invalid Header): 0, Tx Pkt Drops(HPI SAM Overflow): 0

```

[**Jitter Buffer op een gateway configureren**](#)

De QoS-richtlijnen die in dit document worden behandeld, zorgen voor de hausse of de verslechterde spraakqualiteit. De configuratie van de buffer voor de uitloopvertraging is een tijdelijke oplossing voor een ongeschikte QoS-implementatie in het netwerk. Gebruik dit alleen als een lapmiddel of als een middel om problemen op te lossen en de helderheid van de helderder problemen die in het netwerk worden geïntroduceerd te verminderen.

Indelingsmodus

De jitterbuffer is ingesteld voor zowel de vaste modus als de adaptieve modus. Onder de adaptieve modus, kan de poort u een minimum waarde voor de jitter buffer, een maximum waarde en een nominale waarde configureren. De jitter buffer verwacht dat de pakketten binnen het nominale waardebereik zullen aankomen. De nominale waarde moet gelijk zijn aan of meer dan het minimum en gelijk zijn aan of kleiner zijn dan het maximum. De buffer breidt uit tot de ingestelde maximumwaarde. Dit kan tot 1700 msec duren. Eén probleem met het configureren van een hoge maximumwaarde is de introductie van end-to-end vertraging. Kies de waarde van maximum uitloopvertraging zodat dit geen ongewenste vertraging in het netwerk introduceert. Deze uitvoer is een voorbeeld van de jitterbuffer die voor de adaptieve modus is ingesteld:

```
3640-6#
3640-6#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
3640-6(config)#voice-port 3/0/0
3640-6(config-voiceport)#playout-delay mode adaptive
3640-6(config-voiceport)#playout-delay maximum 400
3640-6(config-voiceport)#playout-delay nominal 70
3640-6(config-voiceport)#playout-delay minimum low
3640-6(config-voiceport)#^Z
3640-6#
3640-6#
3640-6#show run | begin 3/0/0
voice-port 3/0/0
  playout-delay maximum 400
  playout-delay nominal 70
  playout-delay minimum low
  playout-delay mode adaptive
!
```

Onder de vaste modus, bekijkt de gateway de ingestelde waarde voor nominaal. Hoewel dit de minimum en de maximum waarde voor de uitloop-vertraging instelt, wordt deze genegeerd wanneer het voor de vaste modus wordt ingesteld. Wanneer in de vaste modus de hoge waterwaarde of de lage waterwaarde altijd constant blijft. Het is gebaseerd op de nominale waarde en is gebaseerd op de waarde van de Rx Delay East (ms). Het is dus mogelijk dat in de vaste modus de waarde 200 msec bedraagt. Als de verwachte vertraging echter dicht bij 100 ms ligt, is dat het hoge watermerk en het lage watermerk voor de gehele duur van de oproep.

```
3640-6#
3640-6#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
3640-6(config)#voice-port 3/0/0
3640-6(config-voiceport)#playout-delay mode fixed
3640-6(config-voiceport)#playout-delay nominal 70
3640-6(config-voiceport)#^Z
3640-6#
3640-6#
3640-6#show run | begin 3/0/0
voice-port 3/0/0
```

```
playout-delay mode fixed
```

```
playout-delay nominal 70
```

```
!
```

Raadpleeg voor meer informatie over de configuratie van de uitzettingsvertraging de [verbeteringen](#) in de [lay-out](#) bij [Voice-over-IP](#).

Gerelateerde informatie

- [Quality-of-Service beleid met DSCP implementeren](#)
- [Low Latency Queueing](#)
- [Het vergelijken van de bandbreedte en de prioriteitsopdrachten van een QoS-servicebeleid](#)
- [Configuratie van Link Fragmentation en Interleaving voor Frame Relay en virtuele ATM-circuits](#)
- [Link Fragmentation and Interleaving configureren voor Multilink PPP](#)
- [Ondersteuning voor spraaktechnologie](#)
- [Productondersteuning voor spraak en Unified Communications](#)
- [Probleemoplossing voor Cisco IP-telefonie](#)
- [Technische ondersteuning - Cisco-systemen](#)