

# Risoluzione dei problemi relativi alle perdite di output con l'accodamento delle priorità

## Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Convenzioni](#)

[Cadute con priorità ip rtp e LLQ](#)

[Interruzioni con coda di priorità legacy](#)

[Misurazione del traffico con un bucket di token](#)

[Procedura di risoluzione dei problemi per diagnosticare le perdite](#)

[Passaggio 1 - Raccolta dati](#)

[Fase 2 - Garantire una larghezza di banda sufficiente](#)

[Fase 3 - Garantire dimensioni burst sufficienti](#)

[Passaggio 4 - Priorità di debug](#)

[Altre cause delle cadute](#)

[Interruzioni code di priorità e Frame Relay](#)

[Informazioni correlate](#)

## [Introduzione](#)

In questo documento vengono forniti suggerimenti per risolvere i problemi di perdite di output derivanti dalla configurazione di un meccanismo di coda di priorità su un'interfaccia del router.

## [Prerequisiti](#)

### [Requisiti](#)

I lettori di questo documento devono conoscere i seguenti concetti:

- **priority-group** o **frame-relay priority-group**: abilita il meccanismo di coda di priorità legacy Cisco. Supporta fino a quattro livelli di code prioritarie.
- **ip rtp priority** o **frame-relay ip rtp priority**: corrisponde ai numeri di porta UDP per il traffico Real-Time Protocol (RTP) che incapsula i pacchetti VoIP e li mette in una coda di priorità.
- **priority**: abilita la funzionalità LLQ (Low Latency Queueing) di Cisco e utilizza la struttura di comando dell'interfaccia della riga di comando (CLI) QoS (Quality of Service) modulare.

Un router può segnalare le perdite di output quando viene configurato uno di questi metodi, ma vi sono importanti differenze funzionali tra i metodi e i motivi delle perdite in ciascun caso.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

## Componenti usati

Il documento può essere consultato per tutte le versioni software o hardware.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

## Convenzioni

Per ulteriori informazioni sulle convenzioni usate, consultare il documento [Convenzioni usate nei suggerimenti tecnici Cisco](#).

## Cadute con priorità ip rtp e LLQ

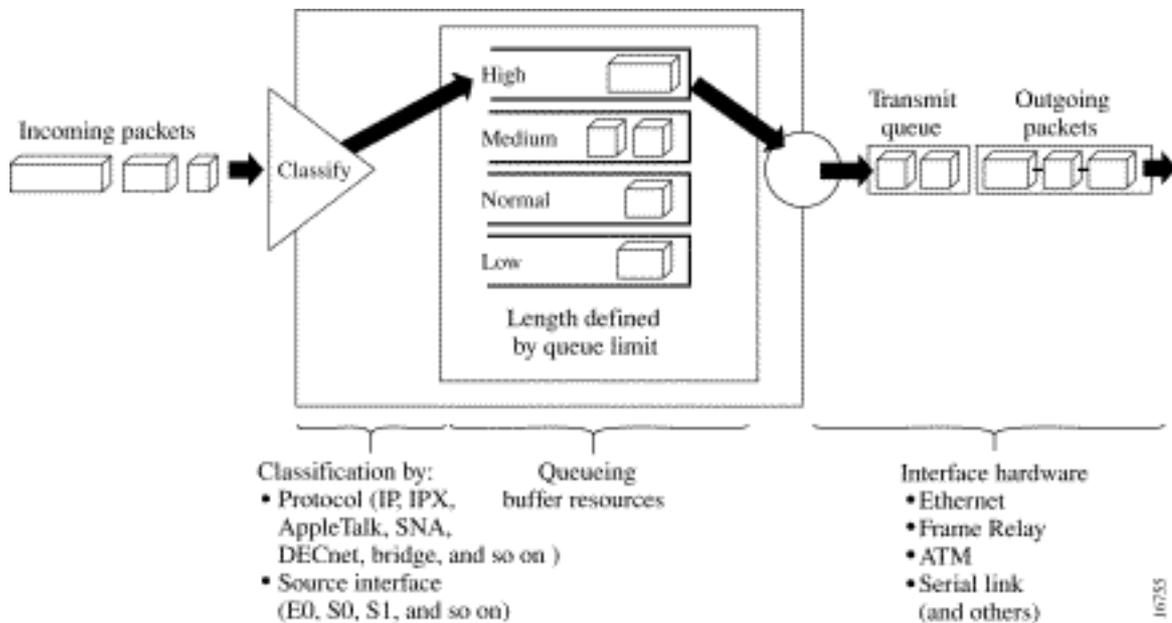
La Guida alla configurazione di Cisco IOS avvisa il sistema di evitare le interruzioni dell'output con questi meccanismi di coda di priorità:

- **priorità ip rtp**: Poiché il comando **ip rtp priority** assegna priorità assoluta rispetto ad altro traffico, deve essere utilizzato con cautela. In caso di congestione, se il traffico supera la larghezza di banda configurata, tutto il traffico in eccesso viene scartato.
- **priority** e LLQ: Quando si specifica il comando **priority** per una classe, viene utilizzato un argomento larghezza di banda che fornisce la larghezza di banda massima. In caso di congestione, il policy viene usato per eliminare i pacchetti quando la larghezza di banda viene superata.

Questi due meccanismi utilizzano un policer integrato per misurare i flussi di traffico. Lo scopo del policer è garantire che le altre code siano servite dall'utilità di pianificazione delle code. Nella funzionalità originale di accodamento priorità di cisco, che utilizza i comandi **priority-group** e **priority-list**, lo scheduler ha sempre servito per primo la coda con priorità più alta. Se c'è sempre stato traffico nella coda ad alta priorità, le code con priorità inferiore sono state private della larghezza di banda e dei pacchetti destinati alle code non prioritarie.

## Interruzioni con coda di priorità legacy

Priority queueing (PQ) è il meccanismo di coda di priorità legacy di cisco. Come illustrato di seguito, PQ supporta fino a quattro livelli di code: alta, media, normale e bassa.



L'attivazione della coda di priorità su un'interfaccia modifica la visualizzazione della coda di output, come illustrato di seguito. Prima della coda di priorità, l'interfaccia Ethernet utilizza una singola coda di attesa di output con la dimensione predefinita di 40 pacchetti.

```
R6-2500# show interface ethernet0
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0000.0c4e.59b1 (bia 0000.0c4e.59b1)
  Internet address is 42.42.42.2/24
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:03, output 00:00:02, output hang never
  Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    239407 packets input, 22644297 bytes, 0 no buffer
    Received 239252 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 input packets with dribble condition detected
    374436 packets output, 31095372 bytes, 0 underruns
    0 output errors, 1 collisions, 13 interface resets
    0 babbles, 0 late collision, 8 deferred
    0 lost carrier, 0 no carrier
    0 output buffer failures, 0 output buffers swapped out
```

Dopo aver abilitato PQ, l'interfaccia Ethernet sta utilizzando quattro code di priorità con limiti di coda variabili, come mostrato nell'output seguente:

```
R6-2500(config)# interface ethernet0
R6-2500(config-if)# priority-group 1
R6-2500(config-if)# end
R6-2500# show interface ethernet 0
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0000.0c4e.59b1 (bia 0000.0c4e.59b1)
  Internet address is 42.42.42.2/24
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
```

```

Last input 00:00:03, output 00:00:03, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: priority-list 1
Output queue (queue priority: size/max/drops):
  high: 0/20/0, medium: 0/40/0, normal: 0/60/0, low: 0/80/0
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  239411 packets input, 22644817 bytes, 0 no buffer
  Received 239256 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  0 input packets with dribble condition detected
  374440 packets output, 31095658 bytes, 0 underruns
  0 output errors, 1 collisions, 14 interface resets
  0 babbles, 0 late collision, 8 deferred
  0 lost carrier, 0 no carrier
  0 output buffer failures, 0 output buffers swapped out

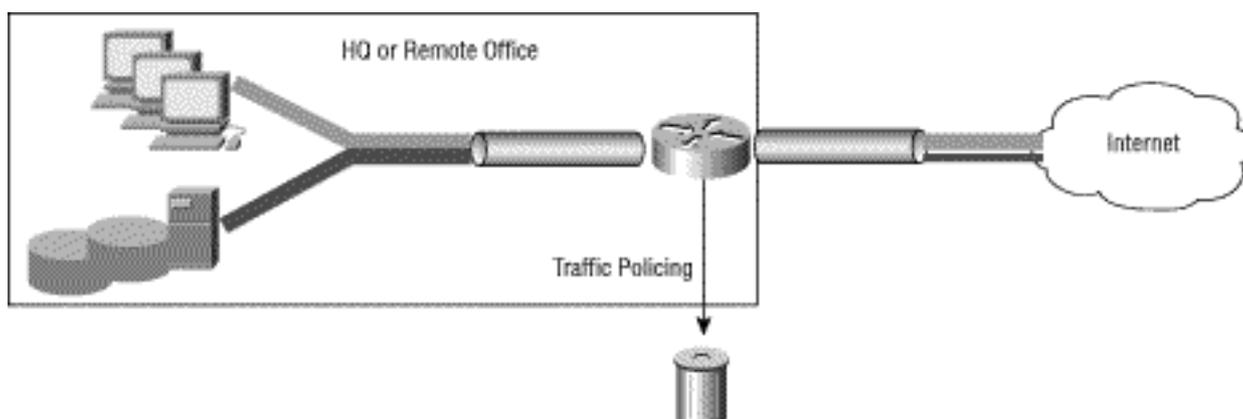
```

Il comando [priority-list {list-number}](#) viene utilizzato per assegnare i flussi di traffico a una coda specifica. Quando i pacchetti arrivano a un'interfaccia, le code di priorità su quell'interfaccia vengono analizzate per trovare i pacchetti in ordine di priorità decrescente. La coda con priorità alta viene analizzata per prima, quindi la coda con priorità media e così via. Il pacchetto all'inizio della coda con priorità più alta viene scelto per la trasmissione. Questa procedura viene ripetuta ogni volta che si invia un pacchetto.

Ogni coda è definita da una lunghezza massima o dal numero massimo di pacchetti che la coda può contenere. Quando un pacchetto in arrivo causa il superamento del limite di coda configurato per la profondità della coda corrente, il pacchetto viene scartato. Pertanto, come indicato in precedenza, le perdite di output con PQ sono in genere dovute al superamento del limite della coda e non a un policer interno, come è il caso tipico di LLQ. Il comando [priority-list list-number queue-limit](#) modifica le dimensioni di una coda di priorità.

## Misurazione del traffico con un bucket di token

Priorità LLQ e IP RTP implementano il policer integrato utilizzando un bucket di token come sistema di misurazione del traffico. In questa sezione viene illustrato il concetto di bucket di token.



Un token bucket non dispone di policy di eliminazione o priorità. La metafora del token bucket è simile alla seguente:

- I token vengono inseriti nel bucket a una determinata velocità.
- Ogni token indica l'autorizzazione per l'origine a inviare un determinato numero di bit nella rete.
- Per inviare un pacchetto, l'autorità di regolamentazione del traffico deve essere in grado di

- rimuovere dal bucket un numero di token corrispondente alle dimensioni del pacchetto.
- Se nel bucket non sono presenti token sufficienti per inviare un pacchetto, il pacchetto attende finché il bucket non dispone di token sufficienti (nel caso di uno shaper) oppure il pacchetto viene scartato o contrassegnato (nel caso di un policer).
  - Il bucket stesso ha una capacità specificata. I token che arrivano dopo che il bucket ha raggiunto la sua capacità massima vengono ignorati e non risultano disponibili per i pacchetti futuri. Pertanto, in qualsiasi momento, la frammentazione più grande che un'applicazione può inviare alla rete è approssimativamente proporzionale alle dimensioni del bucket. Un token bucket consente la burstiness, ovvero l'aumento e la diminuzione intermittenti della frequenza, ma la limita.

Di seguito viene illustrato un esempio relativo all'utilizzo di pacchetti e di una velocità di trasmissione delle informazioni (CIR, Committed Information Rate) di 8000 bps.

- In questo esempio, i bucket di token iniziali iniziano a 1000 byte.
- Quando arriva un pacchetto da 450 byte, il pacchetto è conforme in quanto nel bucket del token conforme sono disponibili byte sufficienti. Il pacchetto viene inviato e 450 byte vengono rimossi dal bucket di token, lasciando 550 byte.
- Quando il pacchetto successivo arriva 0,25 secondi dopo, vengono aggiunti 250 byte al bucket di token  $((0,25 * 8000)/8)$ , lasciando 700 byte nel bucket di token. Se il pacchetto successivo è di 800 byte, il pacchetto supera, e viene scartato. Nessun byte viene preso dal token bucket.

## Procedura di risoluzione dei problemi per diagnosticare le perdite

### Passaggio 1 - Raccolta dati

Di seguito è riportata la procedura per la raccolta dei dati.

1. Eseguire più volte i comandi seguenti e determinare la velocità e la frequenza con cui le perdite si incrementano. Utilizzare l'output per stabilire una base dei modelli di traffico e dei livelli di traffico. Verificare la velocità di rilascio "normale" sull'interfaccia. **show queueing interface**

```
router# show queueing interface hssi 0/0/0
Interface Hssi0/0/0 queueing strategy: priority

Output queue utilization (queue/count)

high/12301 medium/4 normal/98 low/27415
```

**show interface** - Controlla il valore del carico visualizzato nell'output. Inoltre, assicurarsi che la somma dei conteggi delle perdite per coda nell'output **show interface** sia equivalente al conteggio delle perdite di output. Il contatore **show interface** output drops deve visualizzare l'aggregazione totale di tutte le perdite nell'output, inclusi i cali WRED ignorati, i cali scartati a causa di mancanza di buffer (errori "no buffer") e persino i cali nella memoria dell'adattatore di porta integrato.

```
router# show interface serial 4/1/2

Serial4/1/2 is up, line protocol is up
Hardware is cyBus Serial
Description: E1 Link to 60W S9/1/2 Backup
Internet address is 169.127.18.228/27
```

```
MTU 1500 bytes, BW 128 Kbit, DLY 21250 usec, rely 255/255, load 183/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
Last input 00:00:00, output 00:00:00, output hang never
Last clearing of "show interface" counters 5d10h
Input queue: 0/75/0 (size/max/drops); Total output drops: 68277
Queueing strategy: priority-list 7
Output queue: high 0/450/0, medium 0/350/143, normal 0/110/27266, low 0/100/40868
5 minute input rate 959000 bits/sec, 419 packets/sec
5 minute output rate 411000 bits/sec, 150 packets/sec
144067307 packets input, 4261520425 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
42 input errors, 34 CRC, 0 frame, 0 overrun, 1 ignored, 8 abort
69726448 packets output, 2042537282 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 46686454 output buffers swapped out
0 carrier transitions
```

**Nota: alcune interfacce visualizzano valori "txload" e "rxload" separati.**

```
Hssi0/0/0 is up, line protocol is up
Hardware is cyBus HSSI
MTU 1500 bytes, BW 7500 Kbit, DLY 200 usec,
reliability 255/255, txload 138/255, rxload 17/255
Encapsulation FRAME-RELAY IETF, crc 16, loopback not set
Keepalive set (5 sec)
LMI enq sent 4704, LMI stat recvd 4704, LMI upd recvd 0, DTE LMI up
LMI enq recvd 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 1023 LMI type is CISCO frame relay DTE
Broadcast queue 0/256, broadcasts sent/dropped 8827/0, interface
broadcasts 7651
Last input 00:00:00, output 00:00:00, output hang never
Last clearing of "show interface" counters 06:31:58
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 84
Queueing strategy: priority-list 1
Output queue (queue priority: size/max/drops):
high: 0/20/0, medium: 0/40/0, normal: 0/60/0, low: 0/80/84
5 minute input rate 524000 bits/sec, 589 packets/sec
5 minute output rate 4080000 bits/sec, 778 packets/sec
11108487 packets input, 1216363830 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 parity
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
15862186 packets output, 3233772283 bytes, 0 underruns
0 output errors, 0 applique, 1 interface resets
0 output buffer failures, 2590 output buffers swapped out
0 carrier transitions
LC=down CA=up TM=down LB=down TA=up LA=down
```

**show policy-map interface *interface-name*: ricerca di un valore diverso da zero per il contatore "pkts discards".**

```
Router# show policy-map interface s1/0
Serial1/0.1: DLCI 100 -
output : mypolicy
Class voice
  Weighted Fair Queueing
    Strict Priority
    Output Queue: Conversation 72
    Bandwidth 16 (kbps) Packets Matched 0
    (pkts discards/bytes discards) 0/0
Class immediate-data
  Weighted Fair Queueing
    Output Queue: Conversation 73
    Bandwidth 60 (%) Packets Matched 0
    (pkts discards/bytes discards/tail drops) 0/0/0
    mean queue depth: 0
    drops: class random tail min-th max-th mark-prob
```

0	0	0	64	128	1/10
1	0	0	71	128	1/10
2	0	0	78	128	1/10
3	0	0	85	128	1/10
4	0	0	92	128	1/10
5	0	0	99	128	1/10
6	0	0	106	128	1/10
7	0	0	113	128	1/10
rsvp	0	0	120	128	1/10

**Nota:** nell'output di esempio seguente vengono visualizzati i valori corrispondenti per i contatori "packets" e "pkts matched". Questa condizione indica che un numero elevato di pacchetti viene commutato o che l'interfaccia sta sperimentando una congestione estrema. Entrambe queste condizioni possono comportare il superamento del limite di coda di una classe e devono essere esaminate.

```
router# show policy-map interface
```

```
Serial4/0
```

```
Service-policy output: policy1
```

```
Class-map: class1 (match-all)
```

```
189439 packets, 67719268 bytes
```

```
5 minute offered rate 141000 bps, drop rate 0 bps
```

```
Match: access-group name ds-class-af3
```

```
Weighted Fair Queueing
```

```
Output Queue: Conversation 265
```

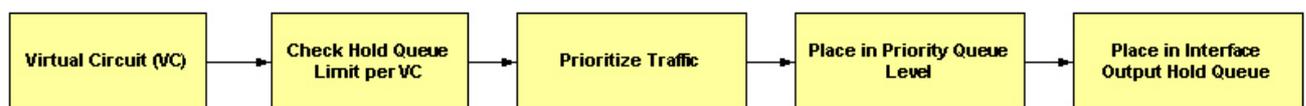
```
Bandwidth 50 (%) Max Threshold 64 (packets)
```

```
(pkts matched/bytes matched) 189439/67719268
```

```
(depth/total drops/no-buffer drops) 0/0/0
```

2. Caratterizzare i flussi di traffico e i pacchetti in essi contenuti. Qual è la dimensione media del pacchetto? In quale direzione scorre il frame con dimensioni MTU? Molti flussi di traffico sono asincroni rispetto al carico. Ad esempio, con un download FTP, la maggior parte dei pacchetti di dimensioni MTU vengono trasferiti dal server FTP al client. I pacchetti dal client FTP al server sono semplici ACK TCP. I pacchetti usano TCP o UDP? Il protocollo TCP consente a ciascun flusso di inviare un numero autorizzato di pacchetti prima che l'origine debba sospendere la trasmissione e attendere che la destinazione riconosca i pacchetti trasmessi.
3. Con Frame Relay, determinare se le perdite si verificano nella coda di interfaccia o in una coda per VC. Il diagramma seguente mostra il flusso dei pacchetti attraverso un circuito virtuale Frame

Relay:



4. Priority Queueing supporta fino a quattro code di output, una per livello di priorità e ogni coda è definita da un limite di coda. Il sistema di coda controlla le dimensioni della coda rispetto al limite di coda configurato prima di inserire il pacchetto in una coda. Se la coda selezionata è piena, il router scarta il pacchetto. Provare ad aumentare le dimensioni della coda con il comando `priority-list {#} queue-limit` e riprendere il monitoraggio.

## [Fase 2 - Garantire una larghezza di banda sufficiente](#)

Con LLQ, il policing consente il trattamento equo di altri pacchetti di dati in altre code CBWFQ (Weighted Fair Queueing) o WFQ basate su classi. Per evitare la perdita dei pacchetti, assicurarsi

di assegnare una quantità ottimale di larghezza di banda alla coda di priorità, tenendo in considerazione il tipo di codec usato e le caratteristiche dell'interfaccia. La priorità IP RTP non consentirà il traffico oltre la quantità allocata.

È sempre più sicuro allocare alla coda delle priorità una quantità di larghezza di banda leggermente superiore a quella richiesta. Si supponga, ad esempio, di aver allocato alla coda di priorità la larghezza di banda di 24 kbps, ovvero la quantità standard necessaria per la trasmissione vocale. Questa allocazione sembra sicura perché la trasmissione di pacchetti voce avviene a una velocità in bit costante. Tuttavia, poiché la rete e il router o lo switch possono utilizzare una parte della larghezza di banda per produrre tremolio e ritardo, l'allocazione di una quantità di larghezza di banda leggermente superiore a quella richiesta (ad esempio 25 kbps) garantisce costanza e disponibilità.

La larghezza di banda allocata per una coda di priorità include sempre l'intestazione di incapsulamento di layer 2. Non include il controllo di ridondanza ciclico (CRC). Per ulteriori informazioni, fare riferimento al documento [Quali byte vengono conteggiati dall'IP per il servizio di coda CoS ATM?](#) per ulteriori informazioni.) Sebbene si tratti solo di pochi byte, il CRC impone un impatto crescente in quanto i flussi di traffico includono un numero maggiore di pacchetti di piccole dimensioni.

Inoltre, sulle interfacce ATM, la larghezza di banda allocata per una coda di priorità non include il seguente sovraccarico fiscale delle celle ATM:

- Qualsiasi spaziatura interna creata dalla segmentazione e dal riassetto (SAR) per rendere l'ultima cella di un pacchetto un multiplo pari di 48 byte.
- CRC a 4 byte del trailer ATM Adaptation Layer 5 (AAL5).
- Intestazione di cella ATM da 5 byte.

Quando si calcola la quantità di larghezza di banda da allocare per una determinata classe di priorità, è necessario tenere conto del fatto che sono incluse le intestazioni di layer 2. Quando si utilizza ATM, è necessario tenere conto del fatto che il sovraccarico fiscale delle celle ATM non è incluso. È inoltre necessario consentire la larghezza di banda per la possibilità di jitter introdotta dai dispositivi di rete nel percorso vocale. Fare riferimento alla sezione [Panoramica delle funzioni di coda a bassa latenza](#).

Quando si utilizza la coda di priorità per trasportare i pacchetti VoIP, fare riferimento al [consumo di larghezza di banda per chiamata Voice over IP](#).

### **Fase 3 - Garantire dimensioni burst sufficienti**

Il trattamento di una serie di pacchetti che lasciano un'interfaccia attraverso una coda di priorità dipende dalle dimensioni del pacchetto e dal numero di byte rimanenti nel bucket del token. È importante considerare le caratteristiche del flusso di traffico diretto alla coda di priorità, in quanto LLQ utilizza un policer, non uno shaper. Un policer utilizza un bucket di token nel modo seguente:

- Il bucket viene riempito con token in base alla frequenza delle classi fino a un massimo del parametro burst.
- Se il numero di token è maggiore o uguale alle dimensioni del pacchetto, il pacchetto viene inviato e il bucket del token viene ridotto. In caso contrario, il pacchetto viene scartato.

Il valore burst predefinito del contatore del traffico del bucket di token di LLQ viene calcolato come 200 millisecondi di traffico alla velocità di larghezza di banda configurata. In alcuni casi, il valore predefinito è inadeguato, in particolare quando il traffico TCP entra nella coda di priorità. I flussi

TCP sono in genere frammentati e possono richiedere dimensioni di frammentazione maggiori di quelle predefinite assegnate dal sistema di coda, in particolare sui collegamenti lenti.

Il seguente output di esempio è stato generato su un PVC ATM con una velocità di cella sostenuta di 128 kbps. Il sistema di coda regola le dimensioni della frammentazione al variare del valore specificato con il comando **priority**.

```
7200-17# show policy-map int atm 4/0.500
ATM4/0.500: VC 1/500 -
```

Service-policy output: drops

```
Class-map: police (match-all)
  0 packets, 0 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
Match: any
Weighted Fair Queueing
  Strict Priority
  Output Queue: Conversation 24
  Bandwidth 90 (%)
  Bandwidth 115 (kbps) Burst 2875 (Bytes)
  !--- Burst value of 2875 bytes is assigned when !--- the reserved bandwidth value is 115
kbps. (pkts matched/bytes matched) 0/0 (total drops/bytes drops) 0/0 Class-map: class-default
(match-any) 0 packets, 0 bytes 5 minute offered rate 0 bps, drop rate 0 bps Match: any 7200-17#
show policy-map int atm 4/0.500
ATM4/0.500: VC 1/500 -
```

Service-policy output: drops

```
Class-map: police (match-all)
  0 packets, 0 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
Match: any
Weighted Fair Queueing
  Strict Priority
  Output Queue: Conversation 24
  Bandwidth 50 (%)
  Bandwidth 64 (kbps) Burst 1600 (Bytes)
  !--- Burst value changes to 1600 bytes when the !--- reserved bandwidth value is changed
to 64 kbps. (pkts matched/bytes matched) 0/0 (total drops/bytes drops) 0/0 Class-map: class-
default (match-any) 0 packets, 0 bytes 5 minute offered rate 0 bps, drop rate 0 bps Match: any
```

La funzionalità di LLQ è stata estesa per consentire una dimensione configurabile Committed Burst (Bc) con la configurazione della dimensione Burst nella funzione di coda a bassa latenza. Grazie a questa nuova funzionalità, la rete è ora in grado di supportare temporanei picchi di traffico e gestire il traffico di rete in modo più efficiente.

Utilizzare il parametro burst con il comando **priority** per aumentare il valore della frammentazione da 1600 byte a 3200 byte.

```
policy-map AV
class AV
priority percent 50 3200
```

**Nota:** un valore elevato aumenta l'effettiva larghezza di banda che la classe di priorità può utilizzare e può dare l'impressione che le classi di priorità ottengano più della giusta quota di larghezza di banda.

Inoltre, il sistema di coda aveva originariamente assegnato un limite di 64 pacchetti alla coda a

bassa latenza. In alcuni casi, quando una frammentazione di 64 pacchetti arriva alla coda di priorità, il contatore del traffico determina che la frammentazione è conforme alla velocità configurata, ma il numero di pacchetti supera il limite della coda. Di conseguenza, alcuni pacchetti sono stati scartati. L'ID bug Cisco [CSCdr51979](#) (solo utenti [registrati](#)) risolve questo problema consentendo alla coda delle priorità di crescere nella profondità consentita dal misuratore del traffico.

Il seguente output è stato acquisito su un PVC Frame Relay configurato con un CIR di 56 kbps. Nel primo set di output campione, la velocità combinata offerta delle classi c1 e c2 è di 76 kbps. Il motivo è che i valori calcolati delle velocità offerte meno le velocità di trasmissione effettive non rappresentano le velocità di trasmissione effettive e non includono i pacchetti che si trovano nello shaper prima di essere trasmessi.

```
router# show policy-map int s2/0.1
Serial2/0.1: DLCI 1000 -

Service-policy output: p

Class-map: c1 (match-all)
  7311 packets, 657990 bytes
  30 second offered rate 68000 bps, drop rate 16000 bps
  Match: ip precedence 1
  Weighted Fair Queueing
    Strict Priority
    Output Queue: Conversation 24
    Bandwidth 90 (%)
    Bandwidth 50 (kbps) Burst 1250 (Bytes)
    (pkts matched/bytes matched) 7311/657990
    (total drops/bytes drops) 2221/199890

Class-map: c2 (match-all)
  7311 packets, 657990 bytes
  30 second offered rate 68000 bps, drop rate 44000 bps
  Match: ip precedence 2
  Weighted Fair Queueing
    Output Queue: Conversation 25
    Bandwidth 10 (%)
    Bandwidth 5 (kbps) Max Threshold 64 (packets)
    (pkts matched/bytes matched) 7310/657900
    (depth/total drops/no-buffer drops) 64/6650/0

Class-map: class-default (match-any)
  2 packets, 382 bytes
  30 second offered rate 0 bps, drop rate 0 bps
  Match: any
```

In questo secondo set di output, i contatori dell'interfaccia **show policy-map** sono stati normalizzati. Sul PVC a 56 kbps, la classe c1 invia circa 50 kbps e la classe c2 circa 6 kbps.

```
router# show policy-map int s2/0.1
Serial2/0.1: DLCI 1000 -

Service-policy output: p

Class-map: c1 (match-all)
  15961 packets, 1436490 bytes
  30 second offered rate 72000 bps, drop rate 21000 bps
  Match: ip precedence 1
  Weighted Fair Queueing
```

```
Strict Priority
Output Queue: Conversation 24
Bandwidth 90 (%)
Bandwidth 50 (kbps) Burst 1250 (Bytes)
(pkts matched/bytes matched) 15961/1436490
(total drops/bytes drops) 4864/437760
```

```
Class-map: c2 (match-all)
 15961 packets, 1436490 bytes
 30 second offered rate 72000 bps, drop rate 66000 bps
Match: ip precedence 2
Weighted Fair Queueing
  Output Queue: Conversation 25
  Bandwidth 10 (%)
  Bandwidth 5 (kbps) Max Threshold 64 (packets)
  (pkts matched/bytes matched) 15960/1436400
  (depth/total drops/no-buffer drops) 64/14591/0
```

```
Class-map: class-default (match-any)
 5 packets, 1096 bytes
 30 second offered rate 0 bps, drop rate 0 bps
Match: any
```

## Passaggio 4 - Priorità di debug

Il comando [debug priority](#) visualizza l'output della coda di priorità se i pacchetti vengono eliminati dalla coda di priorità.

**Attenzione:** Prima di usare i comandi di **debug**, consultare le [informazioni importanti sui comandi di debug](#). Il comando **debug priority** può stampare una grande quantità di output di debug di interruzione su un router di produzione. La quantità dipende dal livello di congestione.

Il seguente output di esempio è stato generato su un Cisco 3640.

```
r3-3640-5# debug priority
Priority output queueing debugging is on

r3-3640-5# ping 10.10.10.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.2, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/57/60 ms
r3-3640-5#
00:42:40: PQ: Serial0/1: ip -> normal
00:42:40: PQ: Serial0/1 output (Pk size/Q 104/2)
00:42:40: PQ: Serial0/1: ip -> normal
00:42:40: PQ: Serial0/1 output (Pk size/Q 104/2)
00:42:40: PQ: Serial0/1: ip -> normal
00:42:40: PQ: Serial0/1 output (Pk size/Q 104/2)
00:42:40: PQ: Serial0/1: ip -> normal
00:42:40: PQ: Serial0/1 output (Pk size/Q 104/2)
00:42:40: PQ: Serial0/1: ip -> normal
00:42:40: PQ: Serial0/1 output (Pk size/Q 104/2)
00:42:41: PQ: Serial0/1 output (Pk size/Q 13/0)
r3-3640-5#no debug priority
00:42:51: PQ: Serial0/1 output (Pk size/Q 13/0)
Priority output queueing debugging is off
```

Nell'output della priorità di **debug** riportato di seguito, 64 indica la profondità effettiva della coda di priorità al momento in cui il pacchetto è stato scartato.

```
*Feb 28 16:46:05.659:WFQ:dropping a packet from the priority queue 64
*Feb 28 16:46:05.671:WFQ:dropping a packet from the priority queue 64
*Feb 28 16:46:05.679:WFQ:dropping a packet from the priority queue 64
*Feb 28 16:46:05.691:WFQ:dropping a packet from the priority queue 64
```

## [Altre cause delle cadute](#)

I seguenti motivi per la perdita di output con LLQ sono stati individuati dal Cisco Technical Assistance Center (TAC) durante la risoluzione dei problemi e documentati in un rapporto sui bug di Cisco:

- L'aumento dei valori di soglia massimi WRED (Weighted Random Early Detection) su un'altra classe ha impoverito i buffer disponibili e ha causato interruzioni nella coda delle priorità. Per facilitare la diagnosi del problema, in una versione futura di IOS è previsto un contatore di "no-buffer drops" per la classe di priorità.
- Se il limite della coda dell'interfaccia di input è inferiore al limite della coda dell'interfaccia di output, il pacchetto viene scaricato e viene spostato sull'interfaccia di input. Questi sintomi sono documentati nell>ID bug Cisco [CSCdu89226](#) (solo utenti [registrati](#)). Risolvere il problema ridimensionando la coda di input e la coda di output in modo appropriato per impedire le interruzioni di input e consentire al meccanismo di coda priorità in uscita di avere effetto.
- L'attivazione di una funzione non supportata nel percorso a commutazione CEF o a commutazione rapida determina la commutazione di contesto di un elevato numero di pacchetti. Con LLQ, i pacchetti a commutazione di contesto sono controllati attualmente indipendentemente dal fatto che l'interfaccia sia congestionata o meno. In altre parole, anche se l'interfaccia non è congestionata, il sistema di coda misura i pacchetti a commutazione di contesto e assicura che il carico offerto non superi il valore della larghezza di banda configurato con il comando **priority**. Questo problema è documentato nell>ID bug Cisco [CSCdv86818](#) (solo utenti [registrati](#)).

## [Interruzioni code di priorità e Frame Relay](#)

Frame Relay è un caso speciale per quanto riguarda il controllo della coda di priorità. La panoramica della funzione [Accodamento a bassa latenza per Frame Relay](#) contiene le seguenti note: "Il PQ è sorvegliato per garantire che le code eque non siano affamate di larghezza di banda. Quando si configura la coda PQ, si specifica in kbps la quantità massima di larghezza di banda disponibile per quella coda. I pacchetti che superano tale limite vengono ignorati." In altre parole, in origine, la coda di priorità di un criterio servizio configurato in una classe di mappa Frame Relay è stata controllata durante i periodi di congestione e non congestione. IOS 12.2 rimuove questa eccezione. PQ è ancora controllato con FRF.12, ma gli altri pacchetti non conformi vengono scartati solo in caso di congestione.

## [Informazioni correlate](#)

- [Pagina di supporto QoS](#)
- [Software Cisco IOS release 12.2 Traffic Policing Feature Module](#)
- [Supporto tecnico – Cisco Systems](#)