

Informazioni su errori e mancati riscontri del buffer

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Convenzioni](#)

[Errori e mancati riscontri buffer](#)

[Pool di buffer](#)

[comando di configurazione buffer](#)

[Comandi aggiuntivi show](#)

[Informazioni correlate](#)

[Introduzione](#)

In questo documento vengono descritti gli errori e gli errori del buffer nel processore di routing (RP).

[Prerequisiti](#)

[Requisiti](#)

Nessun requisito specifico previsto per questo documento.

[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 [Cisco sulle convenzioni nei suggerimenti tecnici](#).

Errori e mancati riscontri buffer

L'RP divide la memoria del processore in pool. Ogni pool contiene un numero di blocchi di memoria di dimensioni uguali. Questi blocchi di memoria sono chiamati buffer.

Pool di buffer

Sono disponibili sei pool di buffer:

- Piccolo—buffer da 104 byte
- Medio: buffer da 600 byte
- Grandi: buffer da 1524 byte
- VeryBig: buffer da 4520 byte
- Grande: buffer da 5024 byte
- Grandissimo: buffer da 18024 byte

Ad esempio, se un processore di interfaccia deve passare un pacchetto da 20 byte all'RP, "chiede" un buffer piccolo. Se un processore di interfaccia deve passare un pacchetto da 500 byte all'RP, chiede un buffer intermedio e così via.

Nota: il processore di interfaccia deve richiedere un buffer di una determinata dimensione.

Quando il processore di interfaccia chiede un buffer, si verifica quanto segue:

- Se il pool richiesto contiene un buffer libero, questo viene concesso. In caso contrario, la richiesta genera un errore e l'algoritmo del buffer tenta di "creare" altri buffer per il pool.
- Quando IOS non riesce a ottenere un buffer piccolo, non scarta il pacchetto. Incrementa il contatore degli errori e passa al buffer di livello successivo, ovvero il buffer intermedio, in cui viene richiesto un buffer. Se non riesce a ottenere un buffer intermedio, richiede il buffer di livello successivo, ovvero un buffer grande. Questo processo continua fino a raggiungere il pool di buffer di grandi dimensioni. Se non riesce a ottenere un buffer di grandi dimensioni, scarta il pacchetto.
- Quando si utilizza il gruppo di funzionalità IBM, una mancata corrispondenza genera quasi sempre un errore.
- Sebbene le funzionalità IBM possano essere commutate in base al processo, il codice per ottenere un buffer per passare un pacchetto da un'interfaccia all'RP viene eseguito a livello di interrupt.
- Non è possibile creare buffer a livello di interrupt; di conseguenza, una richiesta di mancati riscontri accoda la richiesta di ulteriori buffer all'RP.
- Poiché non è possibile creare un buffer aggiuntivo sul posto, la richiesta del buffer ha esito negativo e il pacchetto viene scartato.

Gli errori del buffer sono una delle cause più comuni delle perdite di pacchetti. Quando il pacchetto viene scartato a causa di un errore del buffer, si verifica quanto segue:

- Dopo un errore del buffer, l'RP ha una richiesta in sospeso per creare più buffer delle dimensioni appropriate per il pool specifico.
- Durante la gestione della richiesta di creazione dei buffer da parte dell'RP, è possibile che si verifichino altri errori nel pool.
- L'RP potrebbe addirittura non riuscire a creare più buffer a causa dei vincoli di memoria del

sistema quando sono necessari buffer aggiuntivi.

- In sostanza, l'operazione di creazione dei buffer potrebbe richiedere diversi microsecondi, durante i quali i pacchetti vengono continuamente scartati a causa della mancanza di buffer.
- Inoltre, se i buffer vengono utilizzati con la stessa rapidità con cui vengono creati, l'RP potrebbe essere costretto a dedicare più tempo alla creazione dei buffer che all'elaborazione dei pacchetti.
- Ciò può causare il rilascio dei pacchetti da parte dell'RP così rapidamente da causare una riduzione delle prestazioni e la perdita di sessioni.

Fortunatamente, come si legge in questo documento, i problemi di guasto del buffer non sono difficili da identificare e risolvere. Questo output del comando **show buffers** visualizza lo stato corrente dei pool di buffer del router:

```
dspu-7k#show buffers
```

```
Buffer elements:
```

```
500 in free list (500 max allowed)
2370 hits, 0 misses, 0 created
```

```
Public buffer pools:
```

```
Small buffers, 104 bytes (total 16, permanent 10):
```

```
11 in free list (0 min, 10 max allowed)
1770 hits, 33 misses, 22 trims, 28 created
9 failures (0 no memory)
```

```
Middle buffers, 600 bytes (total 90, permanent 90):
```

```
89 in free list (10 min, 200 max allowed)
590 hits, 0 misses, 0 trims, 0 created
0 failures (0 no memory)
```

```
Big buffers, 1524 bytes (total 90, permanent 90):
```

```
90 in free list (5 min, 300 max allowed)
126 hits, 0 misses, 0 trims, 0 created
0 failures (0 no memory)
```

```
VeryBig buffers, 4520 bytes (total 10, permanent 10):
```

```
10 in free list (0 min, 300 max allowed)
50 hits, 0 misses, 0 trims, 0 created
0 failures (0 no memory)
```

```
Large buffers, 5024 bytes (total 10, permanent 10):
```

```
10 in free list (0 min, 30 max allowed)
0 hits, 0 misses, 0 trims, 0 created
0 failures (0 no memory)
```

```
Huge buffers, 18024 bytes (total 2, permanent 0):
```

```
0 in free list (0 min, 13 max allowed)
2 hits, 2 misses, 0 trims, 2 created
0 failures (0 no memory)
```

Nell'output **show buffers**:

- **Total** identifica il numero totale di buffer nel pool, inclusi i buffer utilizzati e non utilizzati.
- **Permanent** identifica il numero permanente di buffer allocati nel pool. Questi buffer sono sempre nel pool e non possono essere eliminati.
- **In elenco disponibilità** identifica il numero di buffer attualmente presenti nel pool che sono disponibili per l'uso.
- **Min** identifica il numero minimo di buffer che l'RP deve tentare di mantenere nell'elenco di disponibilità: Il parametro **min** viene utilizzato per prevedere la domanda di buffer dal pool in un determinato momento. Se il numero di buffer nell'elenco di disponibilità è inferiore al valore **minimo**, l'RP tenta di creare più buffer per quel pool.
- **Max-allowed** identifica il numero massimo di buffer consentiti nell'elenco di disponibilità: Il

parametro `max-allowed` impedisce a un pool di monopolizzare i buffer non più necessari. Inoltre, libera nuovamente la memoria sul sistema per un ulteriore utilizzo. Se il numero di buffer nell'elenco di disponibilità è maggiore del valore `massimo consentito`, l'RP deve tentare di tagliare i buffer dal pool.

- **Accessi** identifica il numero di buffer richiesti dal pool. Il contatore `accessi` fornisce un meccanismo per determinare quale pool deve soddisfare la richiesta più elevata di buffer.
- **I mancati riscontri** identificano il numero di volte in cui un buffer è stato richiesto e l'RP rilevato in cui sono stati richiesti buffer aggiuntivi del pool. In altre parole, il numero di buffer nell'elenco di disponibilità è sceso sotto il livello `minimo`. Il contatore `Misses` rappresenta il numero di volte in cui l'RP è stato costretto a creare buffer aggiuntivi.
- **Trim** identifica il numero di buffer eliminati dal pool dall'RP, quando il numero di buffer nell'elenco di disponibilità supera il numero di buffer `massimo consentiti`.
- **Creato** identifica il numero di buffer creati nel pool. L'RP crea buffer nelle situazioni seguenti: Quando la richiesta di buffer è aumentata fino a quando il numero di buffer nell'elenco di disponibilità è inferiore al `numero minimo di buffer`. Si verifica un `errore non corretto` perché non sono presenti buffer nell'elenco di disponibilità. Entrambe le situazioni precedenti.
- **Failures** identifica quando IOS non riesce a ottenere un buffer di piccole dimensioni, non elimina il pacchetto. Incrementa il contatore degli errori e passa al buffer di livello successivo, ovvero il buffer intermedio, in cui viene richiesto un buffer. Se non riesce a ottenere un buffer intermedio, richiede il buffer di livello successivo, ovvero un buffer grande. Questo processo continua fino a raggiungere il pool di buffer di grandi dimensioni. Se non riesce a ottenere un buffer di grandi dimensioni, scarta il pacchetto.
- **Nessuna memoria** identifica il numero di errori causati da memoria insufficiente per creare buffer aggiuntivi.

È possibile esaminare le caratteristiche di ogni pool per determinare gli eventuali pool che hanno riscontrato problemi. I parametri di un pool possono essere regolati in modo da consentire al router di essere meglio preparato a gestire il carico, se il pool mostra queste caratteristiche:

- Il numero di accessi non riusciti e crea un incremento a una velocità elevata (come percentuale di accessi riusciti).
- Nell'elenco di disponibilità è presente un numero costantemente basso di buffer.
- Numero di `errori` o `nessun` incremento di `memoria`.

[comando di configurazione buffer](#)

Il comando di configurazione dei **buffer** consente di regolare i seguenti parametri per ciascun pool di buffer:

- **initial** - Buffer temporanei allocati al ricaricamento del sistema.
- **max-free** - Numero massimo di buffer liberi.
- **min-free** - Numero minimo di buffer liberi.
- **permanent** - Numero di buffer permanenti.

[Buffer iniziali](#)

Sintonizzare i buffer **iniziali** per gestire la frammentazione del traffico di istituzione della sessione dopo il ricaricamento del router.

```
buffers small initial 250
```

Questi buffer vengono alla fine "tagliati" e restituiti al sistema.

I buffer iniziali sono progettati per gestire l'impostazione della sessione, che è sempre a commutazione di contesto.

Durante la creazione della sessione, la cache fastswitching (utilizzata da altri protocolli di routing) viene popolata; i buffer a commutazione di contesto non sono più necessari e possono essere restituiti al sistema.

Il tuning dei buffer iniziali potrebbe non essere la soluzione corretta per il set di funzionalità IBM, in quanto quasi tutti i pacchetti (dopo l'istituzione della sessione) sono commutati in base al processo e richiedono comunque il buffer aggiuntivo.

Nota: per le funzioni IBM a commutazione di contesto, è necessario sintonizzare i buffer **permanenti** anziché quelli temporanei iniziali.

[Numero massimo buffer liberi](#)

Sintonizzare i buffer **max-free** in modo che il valore sia uguale o maggiore dei buffer permanenti. Se tutti i buffer permanenti sono nell'elenco di disponibilità, l'RP non deve tentare di eliminare i buffer permanenti. Max-free può essere utilizzato per garantire che i buffer inutilizzati creati durante i burst irregolari vengano restituiti alla memoria di sistema.

```
buffers small max-free 175
```

```
buffers small permanent 125
```

[Buffer liberi minimi](#)

Ottimizzare i buffer **privi di min** in modo che il valore rappresenti il numero minimo stimato di buffer richiesti in qualsiasi momento. Min-free può essere utilizzato per anticipare le condizioni di carenza di buffer e per garantire che sia sempre disponibile un numero minimo di buffer.

```
buffers small min-free 50
```

[Buffer permanenti](#)

Ottimizzare i buffer **permanenti** in modo che il valore rappresenti il numero stimato di buffer necessari per la normale elaborazione.

```
buffers small permanent 125
```

I buffer permanenti vengono utilizzati per soddisfare i normali requisiti del buffer (compresi i burst frequenti) del router. La determinazione dei normali requisiti del buffer è un processo interattivo, in cui l'output **show buffer** deve visualizzare il totale dei buffer utilizzati in un pool in un determinato momento. I buffer permanenti devono essere sintonizzati in relazione ai buffer "totali" coerenti richiesti. Quando si sintonizzano i buffer permanenti, è necessario concentrarsi sulla riduzione delle creazioni e sull'eliminazione degli errori e dei mancati riscontri.

Comandi aggiuntivi show

Per identificare i problemi relativi all'allocazione dei buffer, è possibile usare altri due comandi **show**:

- **show interfaces *interface-identifier***
- **show source-bridge**

Questo output del comando di esempio **show interfaces *interface-identifier*** include un contatore per l'assenza di buffer:

```
dspu-7k#show interfaces channel 4/2
```

```
Channel4/2 is up, line protocol is up
Hardware is cxBus IBM Channel
MTU 4472 bytes, BW 98304 Kbit, DLY 100 usec, rely 255/255, load 1/255
Encapsulation CHANNEL, loopback not set, keepalive not set
Virtual interface
Last input 0:00:04, output 0:00:04, output hang never
Last clearing of "show interface" counters never
Output queue 0/40, 0 drops; input queue 0/75, 8 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  646 packets input, 27760 bytes, 8 no buffer
  Received 0 broadcasts, 0 runts, 0 giants
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  328 packets output, 16959 bytes, 0 underruns
  0 output errors, 0 collisions, 0 interface resets, 0 restarts
  0 output buffer failures, 0 output buffers swapped out
```

Nell'output del comando **show interfaces *interface-identifier***.

- Il contatore `no buffer` viene incrementato quando l'interfaccia non riesce a ottenere un buffer per un pacchetto in entrata.
- I contatori `no buffer` e `drop` (coda di input) vengono incrementati quando l'interfaccia non riesce a ottenere un buffer per un pacchetto in entrata.
- Un contatore `no buffer` incrementato nell'output **show interfaces** è correlato al contatore `missing` incrementato nell'output **show buffers**. È possibile ottimizzare il pool di buffer appropriato.

Questo output del comando di esempio **show source-bridge** include un contatore di interfaccia per la velocità, quando il bridging source-route (SRB) è configurato per l'interfaccia:

```
dspu-7k#show source-bridge
```

Local Interfaces:		receive	transmit				
srn	bn	trn	r p s n	max hops	cnt:bytes	cnt:bytes	drops
Ch4/2	666	1	99 * f	7 7 7	652:26020	6:266	0

Global RSRB Parameters:

TCP Queue Length maximum: 100

Ring Group 99:

This TCP peer: 150.10.20.2

Maximum output TCP queue length, per peer: 100

Peers:	state	bg	lv	pkts_rx	pkts_tx	expl_gn	drops	TCP
TCP 150.10.20.1	open		*3	261	266	0	0	0
TCP 150.10.20.2	-		*3	0	0	0	0	0

Rings:

bn: 1	rn: 888	locvrt	ma: 4000.7000.fff1	Buff Ring888		fwd: 0
bn: 1	RN: 666	local	ma: 4000.0c48.2e80	Channel4/2		fwd: 261
bn: 1	RN: 88	remote	ma: 4000.4000.fff1	TCP 150.10.20.1		fwd: 322
bn: 1	RN: 250	remote	ma: 4000.300f.7c09	TCP 150.10.20.1		fwd: 0

Explorers:	input			output		
	spanning	all-rings	total	spanning	all-rings	total
Ch4/2	0	0	0	0	1	1

Local: fastswitched 0 flushed 0 max Bps 256000

rings	inputs	bursts	throttles	output	drops
Ch4/2	0	0	8		0

Nell'output del comando **show source-bridge**:

- Il contatore di `velocità` si incrementa quando l'interfaccia non riesce a ottenere un buffer per un pacchetto in entrata.
- Il contatore `throttles` che aumenta nell'output del comando **show interfaces** è correlato a un contatore `misses` che aumenta nell'output del comando **show buffers**. È possibile ottimizzare il pool di buffer appropriato.

[Informazioni correlate](#)

- [Sintonizzazione buffer per tutti i router Cisco](#)
- [Documentazione e supporto tecnico – Cisco Systems](#)