

Solución de problemas de vaciado en la interfaz

Contenido

[Introducción](#)

[Antecedentes](#)

[Problema: Flushes Drop on the Interface](#)

Introducción

Este documento describe cómo resolver problemas de vaciados en la interfaz cuando se toma el resultado del comando show interfaces en el router.

Antecedentes

Los vaciados se utilizan para contar el descarte selectivo de paquetes (SPD). Es un mecanismo que descarta rápidamente los paquetes de baja prioridad cuando la CPU está sobrecargada para guardar cierta capacidad de procesamiento para los paquetes de alta prioridad. El contador de purga en el resultado de comando show interface se incrementa como parte del descarte de paquetes selectivos (SPD), que implementa una política para descartar paquetes selectivos en la cola del proceso del IP del router. Por lo tanto, se aplica al único tráfico conmutado por proceso.

El propósito del SPD es asegurarse de que los paquetes de control importantes, como actualizaciones de ruteo y keepalives, no se descarten cuando la cola de entrada del IP esté llena. Cuando el tamaño de la cola de entrada del IP se encuentre entre los umbrales mínimos y máximos, se descartan los paquetes normales del IP en función de cierta probabilidad de descarte. Este descarte al azar se denomina purga SPD.

Problema: Flushes Drop on the Interface

Las caídas de los vaciados pueden causar inaccesibilidad, lentitud, problemas de degradación de calidad en el link. Puede verificar el contador de purga con este comando en el router.

```
Router# Show interface GigabitEthernet 0/0
```

```
GigabitEthernet0/0/0 is administratively down, line protocol is down
  Hardware is BUILT-IN-2T+6X1GE, address is 0035.1a53.7302 (bia 0035.1a53.7302)
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive not supported
  Full Duplex, 1000Mbps, link type is auto, media type is T
  output flow-control is on, input flow-control is on
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/14323 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
```

```

5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts (0 IP multicasts)
  0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
  0 watchdog, 0 multicast, 0 pause input
  0 packets output, 0 bytes, 0 underruns
  0 output errors, 0 collisions, 3 interface resets
  0 unknown protocol drops
  0 babbles, 0 late collision, 0 deferred
  0 lost carrier, 0 no carrier, 0 pause output
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions

```

Estos vaciados se pueden ver incluso cuando no hay congestión en el link. El incremento en la cola de espera también puede no resolver el problema en tales casos. A veces, los vaciados son buenos ya que descartan el tráfico de baja prioridad sobre el tráfico prioritario. Si ve los vaciados en la interfaz que son intermitentes, entonces este script se puede utilizar para obtener la información sobre el tráfico que está atascado en los búfers de la interfaz. El script EEM aquí se crea con GigabitEthernet0/0 mantenido como la interfaz afectada. Las modificaciones se pueden realizar de acuerdo con la interfaz en la que desea resolver problemas y el tamaño máximo de cola establecido. Aparte de esto, el valor de 74 es el valor máximo predeterminado para iniciar la caída completa cuando el tamaño de cola se especifica como 75. Puede establecer manualmente el umbral mínimo y máximo con el comando mencionado al final del documento junto con los detalles de estas caídas.

```

event manager applet input_queue_watch
 event timer watchdog time 5
 action 1.0 cli command "enable"
 action 2.0 cli command "show interface GigabitEthernet0/0 | inc Input queue"
 action 3.0 regexp "Input queue: ([0-9]+)/75/" $_cli_result match qsize
 action 4.0 if $_regexp_result eq 1
 action 4.1  if $qsize ge 74
 action 4.2  cli command "term exec prompt time"
 action 4.3  cli command "show ip traffic | append flash:queue_log.log"
 action 4.4  cli command "show ip cef not | append flash:queue_log.log"
 action 4.5  cli command "show ip cef switching state | append flash:queue_log.log"
 action 4.6  cli command "show buffer input-interface GigabitEthernet0/0 packet | append
flash:queue_log.log"
 action 4.7  cli command "show buffer input-interface GigabitEthernet0/0 header | append
flash:queue_log.log"
 action 4.8  end

```

El resultado de **Show buffer input-interface GigabitEthernet0/0** y **Show buffer input-interface GigabitEthernet0/0 packet** le da la información de tráfico que está allí en la cola.

```

Router# Show buffer input-interface fa0/0

```

Header	DataArea	Pool	Rcnt	Size	Link	Enc	Flags	Input	Output
64C22054	DA00084	Small	1	62	7	1	200	Fa0/0	None
64C238B8	DA00944	Small	1	62	7	1	200	Fa0/0	None
64C24A24	DA00F84	Small	1	62	7	1	200	Fa0/0	None
64C2511C	DA01204	Small	1	62	7	1	200	Fa0/0	None
64C25814	DA01484	Small	1	62	7	1	200	Fa0/0	None
64C26288	DA01844	Small	1	62	7	1	200	Fa0/0	None
64C26CFC	DA01C04	Small	1	91	7	1	280	Fa0/0	None
64C27078	DA01D44	Small	1	62	7	1	200	Fa0/0	None
64C273F4	DA01E84	Small	1	62	7	1	200	Fa0/0	None

65251C34	DD1F024	Small	1	62	7	1	200	Fa0/0	None
653A54B8	DD1FF24	Small	1	62	7	1	200	Fa0/0	None
653A5834	DD20064	Small	1	62	7	1	200	Fa0/0	None
653A69A0	DD206A4	Small	1	62	7	1	200	Fa0/0	None
6542C338	DD1FB64	Small	1	62	7	1	200	Fa0/0	None

Router# Show buffer input-interface GigabitEthernet0/0 packet

```

Buffer information for Small buffer at 0x64C25498
  data_area 0xDA01344, refcount 1, next 0x0, flags 0x200
  linktype 7 (IP), enctype 1 (ARPA), encsize 14, rxtype 1
  if_input 0x64F2391C (FastEthernet0/0), if_output 0x0 (None)
  inputtime 15:45:44.284 (elapsed 00:00:02.956)
  outputtime 00:00:00.000 (elapsed never), oqnumber 65535
  datagramstart 0xDA0138A, datagramsize 62, maximum size 260
  mac_start 0xDA0138A, addr_start 0xDA0138A, info_start 0x0
  network_start 0xDA01398, transport_start 0xDA013AC, caller_pc 0x6072308C

```

```

source: 172.18.162.125, destination: 172.18.13.175, id: 0x47C6, ttl: 1,
TOS: 0 prot: 6, source port 1433, destination port 1390:

```

```

....
  4: 62800030 85142082 08004500 003037AA b..0.. ..E..07*
 20: 40000106 409FAC12 A56DAC12 03ED044F @...@.,.%m,..m.O
 36: 008B9D84 24630000 00007002 80003ADE ....$c....p....^
 52: 00000204 05B40101 040200 .....4.....

```

```

Buffer information for Small buffer at 0x64C24DA0
  data_area 0xDA010C4, refcount 1, next 0x65246DC0, flags 0x200
  linktype 7 (IP), enctype 1 (ARPA), encsize 14, rxtype 1
  if_input 0x64F2391C (FastEthernet0/0), if_output 0x0 (None)
  inputtime 15:45:41.944 (elapsed 00:00:00.056)
  outputtime 00:00:00.000 (elapsed never), oqnumber 65535
  datagramstart 0xDA0110A, datagramsize 62, maximum size 260
  mac_start 0xDA0110A, addr_start 0xDA0110A, info_start 0x0
  network_start 0xDA01118, transport_start 0xDA0112C, caller_pc 0x6072308C

```

```

source: 172.18.162.115, destination: 172.18.71.102, id: 0xC58F, ttl: 1,
TOS: 0 prot: 6, source port 4952, destination port 139

```

```

  0: 00078509 62800030 ....b..0
  8: 85142082 08004500 0030C58F 40000106 .. ..E..0E.@...
 24: 723AAC12 A273AC12 47661358 008B013D r:,. "s,.Gf.X...=
 40: 71660000 00007002 80003A9A 00000204 qf....p....:.....
 56: 05B40101 040200

```

```

Buffer information for Small buffer at 0x64C22054
  data_area 0xDA00084, refcount 1, next 0x653A62A8, flags 0x200
  linktype 7 (IP), enctype 1 (ARPA), encsize 14, rxtype 1
  if_input 0x64F2391C (FastEthernet0/0), if_output 0x0 (None)
  inputtime 15:45:34.756 (elapsed 00:00:05.348)
  outputtime 00:00:00.000 (elapsed never), oqnumber 65535
  datagramstart 0xDA000CA, datagramsize 62, maximum size 260
  mac_start 0xDA000CA, addr_start 0xDA000CA, info_start 0x0
  network_start 0xDA000D8, transport_start 0xDA000EC, caller_pc 0x6072308C

```

```

source: 172.18.100.7, destination: 172.18.101.147, id: 0x684A, ttl: 255, prot: 1

```

```

  0: 00078509 62800030 85142082 08004500 ....b..0.. ..E.
 16: 0030C32E 40000106 2589AC12 A273AC12 .OC.@...%.,."s,.
 32: 967811E6 01BD1253 53C40000 00007002 .x.f.=.SSD....p.
 48: 8000F853 00000204 05B40101 040200 ..xS.....4.....

```

```

Buffer information for Small buffer at 0x64B7C588

```

```
data_area 0xDDA5484, refcount 1, next 0x65DC5D8C, flags 0x200
linktype 7 (IP), encctype 1 (ARPA), encsize 14, rxtype 1
if_input 0x64F2391C (FastEthernet0/0), if_output 0x0 (None)
inputtime 15:45:21.408 (elapsed 00:00:00.300)
outputtime 00:00:00.000 (elapsed never), oqnumber 65535
datagramstart 0xDDA54CA, datagramsize 62, maximum size 260
mac_start 0xDDA54CA, addr_start 0xDDA54CA, info_start 0x0
network_start 0xDDA54D8, transport_start 0xDDA54EC, caller_pc 0x6072308C
```

```
source: 172.18.101.147, destination: 172.18.246.99, id: 0x3BE6, ttl: 1,
TOS: 0 prot: 6, source port 3096, destination port 139
```

```
0: 00078509 62800030      ....b..0
8: 85142082 08004500 00303BE6 40000106  .. ...E..0;f@...
24: 89C6AC12 6593AC12 F6630C18 008BBEB1  .F,.e.,.vc....>1
40: 4A500000 00007002 8000395E 00000204  JP....p...9^....
56: 05B40101 040200      .4.....
```

```
Buffer information for Small buffer at 0x64C24DA0
```

```
data_area 0xDA010C4, refcount 1, next 0x653A6D1C, flags 0x200
linktype 7 (IP), encctype 1 (ARPA), encsize 14, rxtype 1
if_input 0x64F2391C (FastEthernet0/0), if_output 0x0 (None)
inputtime 15:45:17.192 (elapsed 00:00:00.028)
outputtime 00:00:00.000 (elapsed never), oqnumber 65535
datagramstart 0xDA0110A, datagramsize 62, maximum size 260
mac_start 0xDA0110A, addr_start 0xDA0110A, info_start 0x0
network_start 0xDA01118, transport_start 0xDA0112C, caller_pc 0x6072308C
```

```
source: 172.18.165.109, destination: 172.18.149.166, id: 0x28BC, ttl: 1,
TOS: 0 prot: 6, source port 4086, destination port 445
```

```
0: 00078509      ....
4: 62800030 85142082 08004500 003028BC  b..0.. ...E..0(<
20: 40000106 BDD3AC12 A56DAC12 95A60FF6  @...=S,.%m,..&.v
36: 01BD9A3D 72370000 00007002 800051BE  .=.=r7....p...Q>
52: 00000204 05B40101 040200      .....4.....
```

```
Buffer information for Small buffer at 0x653A6624
```

```
data_area 0xDD20564, refcount 1, next 0x65343F50, flags 0x200
linktype 7 (IP), encctype 1 (ARPA), encsize 14, rxtype 1
if_input 0x64F2391C (FastEthernet0/0), if_output 0x0 (None)
inputtime 15:46:12.888 (elapsed 00:00:00.012)
outputtime 00:00:00.000 (elapsed never), oqnumber 65535
datagramstart 0xDD205AA, datagramsize 62, maximum size 260
mac_start 0xDD205AA, addr_start 0xDD205AA, info_start 0x0
network_start 0xDD205B8, transport_start 0xDD205CC, caller_pc 0x6072308C
```

```
source: 172.18.165.109, destination: 172.18.159.108, id: 0x4902, ttl: 1,
TOS: 0 prot: 6, source port 2391, destination port 445
```

```
0: 00078509 62800030 85142082 08004500  ....b..0.. ...E.
16: 00304902 40000106 93C7AC12 A56DAC12  .0I.@....G,.%m,.
32: 9F6C0957 01BDA1C0 C57C0000 00007002  .l.W.=!@E|...p.
48: 8000F3CE 00000204 05B40101 040200      ..sN.....4.....
```

Una vez que descubra el tráfico que se encuentra en cola, puede tomar las acciones necesarias sobre él. Podría ser la limitación de velocidad de ese tráfico o si el tráfico no es legítimo, puede aplicar una ACL para bloquear el tráfico.

Si el contador aumenta al solucionar problemas, también puede ejecutar los comandos manualmente. Tenga en cuenta que los comandos **Show buffer input-interface GigabitEthernet0/0** y **Show buffer input-interface GigabitEthernet0/0 packet** a veces no dan el resultado a la vez, por lo que puede que necesite ejecutar el comando un par de veces.

La cola de proceso en el RP se divide en dos partes: una cola de paquetes general y una cola de prioridad. Los paquetes colocados en la cola de paquetes general están sujetos a la verificación de estado SPD, y los que se ponen en la cola de prioridad no lo están. Los paquetes que califican para la cola de paquetes de prioridad son paquetes de alta prioridad como los de precedencia IP 6 o paquetes IGP y nunca deben descartarse. Sin embargo, los paquetes que no califican pueden descartarse aquí, según la longitud de la cola general de paquete en función del estado SPD. La cola general de paquetes puede estar en tres estados y, como tal, los paquetes de baja prioridad pueden recibir un servicio diferente:

- NORMAL: tamaño de cola \leq min
- DESCARTE ALEATORIO: min \leq tamaño de cola \leq máx.
- CAÍDA COMPLETA: max \leq tamaño de cola

En el estado NORMAL, los paquetes bien formados y mal formados nunca se descartan. En el estado DROP ALEATORIO, los paquetes bien formados se descartan aleatoriamente. Si se configura el modo agresivo, se descartan todos los paquetes mal formados; de lo contrario, los tratamos como paquetes bien formados. En el estado FULL DROP, se descartan todos los paquetes bien formados y mal formados. Estos valores min (valor predeterminado 73) y max (valor predeterminado 74) se derivan de la cola en espera más pequeña del chasis, pero se pueden reemplazar con los comandos globales **ip spd queue min-threshold** y **ip spd queue max-threshold**.