故障排除由 IP Input 进程引起的 High CPU 问题

目录

[问题描述](#)
[故障诊断步骤](#)

问题描述

我们经常会遇到High CPU的问题，一般来说CPU 分进程高和中断高两种。中断高很多时候是由于实时的流量造成的，要根据网络的实际情况具体分析，原因有很多种，在此不一一列举了。进程高，不同的进程也有不同的root cause。 在比较常见的几种占用CPU资源进程中，有一种叫 IP Input。

在 IOS 中，我们把 SW process 叫做 IP Input 进程，顾名思义就是数据报文没有被硬件 switching cache 或者 CEF 处理，而是 punt 到 CPU 去做进一步的处理。对于一个数据包而言，有几种情况会被 punt 到 CPU。

1. 在 switching cache 里没有该报文相关的条目。如该报文的目的地址是 1.1.1.1 而这个目的前缀在如接口所在的 cache 里找不到该条目，那么这个报文将上送CPU做进一步查找。
2. 目的地址是到本机的报文。
3. 广播报文。
4. IP 头里携带 option 的报文。
5. 需要协议转换的报文。
6. 需要加密或者压缩的报文（如果有了 CSA(Compression Service Adapter)和 ESA(Encryption Service Adapter) 则不需要上送 CPU 只是本地处理了）。
7. 分片报文或者需要重组的报文。（MTU设置的不合理，就可能会出发这类问题）。

故障诊断步骤

步骤 1，通过 **show cpu process** 或者 **sh processes cpu sorted | exclude 0.00** 来查看是 CPU 高，CPU 资源是否是被 IP Input 或者其他资源占用了。

```
 ln-t1a-rt05>sh processes cpu sorted | exclude 0.00
CPU utilization for five seconds: 97%/40%; one minute: 99%; five minutes: 99%
 PID Runtime(ms)   Invoked      uSecs   5Sec   1Min   5Min TTY Process
 188   112780212 404873385        278 54.14% 47.34% 48.09%   0 IP Input
 452    32291128 170096816        189  1.43%  0.85%  0.60%   0 BGP Router
 444    31584160  54483137        579  0.31%  0.19%  0.17%   0 SNMP ENGINE
 442     7514572  54233626        138  0.15%  0.12%  0.10%   0 IP SNMP
   2      239608   2221363        107  0.07%  0.06%  0.05%   0 Load Meter
 332        9940      2296       4329  0.07%  0.07%  0.05%   0 Syslog Traps
 433    14477844   2116934       6839  0.07%  0.13%  0.16%   0 Tag Control
 321    12935848 171140132         75  0.07%  0.04%  0.05%   0 BGP I/O
 259      660104  50553708         13  0.07%  0.04%  0.05%   0 TCP Timer
 264      417244   1429570        291  0.07%  0.62%  0.54%   0 XDR mcast
 446      265680    424691        625  0.07%  0.19%  0.22%   0 SNMP Traps
```

步骤 2，如果是 IP Input 进程导致的 CPU 利用率高，那么执行 **show interface status** 和 **show interface**，来查看哪些接口有较多的流量需要转发，以及这些接口是用何种转发机制，(fast

switching / cef switching /processor)

```
sh int stat

Switching path    Pkts In   Chars In   Pkts Out  Chars Out
              Processor     650856   73671257     526830   62471991
            Route cache  330217560  821433561    6624091 1799996569
      Distributed cache          0          0          0          0
                  Total  330868416  895104818    7150921 1862468560
Interface POS0/1/0 is disabled  <<<<<<<<

FastEthernet1/0/0
         Switching path    Pkts In   Chars In   Pkts Out  Chars Out
              Processor  376139751  259046650  373525823   58361131
            Route cache  212836164 1931396850  344019086  233164671
      Distributed cache          0          0          0          0
                  Total  588975961 2190447507  717544911  291525922
Serial1/1/0
         Switching path    Pkts In   Chars In   Pkts Out  Chars Out
              Processor      34782    9898739     869284   85447616
            Route cache   14178391 3542542743  206212532 2364427002
      Distributed cache          0          0          0          0
                  Total   14213173 3552441482  207081816 2449874618
Commtouch-SC4-7513#
```

步骤 3，**show ip traffic** 的输出，可以告诉我们是哪类流量增长最快，之后我们再检查一下这类流量是否需要上送 CPU 做进一步处理的（参考上述第一部分）。就能得出一个大致的结论----High CPU 问题是什么流量导致的。

```
Commtouch-SC4-7513#sh ip traffic
IP statistics:
 Rcvd:  3764825155 total, 1632039 local destination
        0 format errors, 0 checksum errors, 46716 bad hop count
        0 unknown protocol, 0 not a gateway
        0 security failures, 0 bad options, 7772 with options
 Opts:  103 end, 10 nop, 387 basic security, 0 loose source route
        0 timestamp, 0 extended security, 44 record route
        0 stream ID, 0 strict source route, 7282 alert, 0 cipso
        0 other
 Frags: 0 reassembled, 0 timeouts, 0 couldn't reassemble
        4 fragmented, 0 couldn't fragment
 Bcast: 103181 received, 61 sent
 Mcast: 0 received, 32678 sent
 Sent:  2929975 generated, 3758313272 forwarded
 Drop:  54546 encapsulation failed, 0 unresolved, 0 no adjacency
        589 no route, 0 unicast RPF, 0 forced drop

ICMP statistics:
 Rcvd: 0 format errors, 0 checksum errors, 0 redirects, 10299
unreachable
       319787 echo, 24545 echo reply, 0 mask requests, 0 mask replies,
1
quench
```

```
       0 parameter, 0 timestamp, 0 info request, 0 other
       0 irdp solicitations, 0 irdp advertisements
 Sent: 0 redirects, 1205238 unreachable, 24671 echo, 319787 echo reply
       0 mask requests, 0 mask replies, 0 quench, 0 timestamp
       0 info reply, 45869 time exceeded, 0 parameter problem
       0 irdp solicitations, 0 irdp advertisements

UDP statistics:
 Rcvd: 390403 total, 1 checksum errors, 292907 no port
 Sent: 199023 total, 5 forwarded broadcasts

TCP statistics:
 Rcvd: 883671 total, 0 checksum errors, 1878 no port
 Sent: 1135440 total

Probe statistics:
 Rcvd: 0 address requests, 0 address replies
       0 proxy name requests, 0 where-is requests, 0 other
 Sent: 0 address requests, 0 address replies (0 proxy)
       0 proxy name replies, 0 where-is replies

EGP statistics:
 Rcvd: 0 total, 0 format errors, 0 checksum errors, 0 no listener
 Sent: 0 total

IGRP statistics:
 Rcvd: 0 total, 0 checksum errors
 Sent: 0 total

OSPF statistics:
 Rcvd: 0 total, 0 checksum errors
       0 hello, 0 database desc, 0 link state req
       0 link state updates, 0 link state acks

 Sent: 0 total

IP-IGRP2 statistics:
 Rcvd: 0 total
 Sent: 0 total

PIMv2 statistics: Sent/Received
 Total: 0/0, 0 checksum errors, 0 format errors
 Registers: 0/0, Register Stops: 0/0,  Hellos: 0/0
 Join/Prunes: 0/0, Asserts: 0/0, grafts: 0/0
 Bootstraps: 0/0, Candidate_RP_Advertisements: 0/0

IGMP statistics: Sent/Received
 Total: 0/0, Format errors: 0/0, Checksum errors: 0/0
 Host Queries: 0/0, Host Reports: 0/0, Host Leaves: 00
 DVMRP: 0/0, PIM: 0/0

ARP statistics:
 Rcvd: 2159586 requests, 978 replies, 0 reverse, 0 other
```

```
 Sent: 8855 requests, 27586 replies (25987 proxy), 0 reverse

==========
Commtouch-SC4-7513#sh tech
```

当然，我们也可以通过 **debug ip packet(detail)** 来更直接地查看到底是什么样的报文上送到CPU触发了CPU High，但是由于此时CPU的利用率已经很高了，所以建议先执行

```
router(config)#no logging console
router(config)#no logging monitor
```

之后再执行 logging buffered，这样 Debug 的结果将被直接记录到 log buffer 里，而不会再 session 里不停的弹出。最后，**show log** 就可以看到具体的报文了。

现在可以开始 debug:

```
router#debug ip packet detail
IP packet debugging is on (detailed)
```

Debug最长不能超过 3 到 5 秒钟。可以使用 undebug all 命令停止：

```
router#undebug all
All possible debugging has been turned off
```

可以使用show logging 命令检查结果：

```
router#show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: disabled
    Monitor logging: disabled
    Buffer logging: level debugging, 145 messages logged
    Trap logging: level informational, 61 message lines logged
Log Buffer (64000 bytes):

*Mar  3 03:43:27.320: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.204
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.324: ICMP type=8, code=0
*Mar  3 03:43:27.324: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.205
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.324: ICMP type=8, code=0
*Mar  3 03:43:27.328: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.206
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.328: ICMP type=8, code=0
...
```

该log显示：

- 每 4ms 收到一个包
- 报文的源 IP 地址是 192.168.40.53
- 报文从 interface Ethernet0/1 进入的.
- 这些上送 CPU 的报文有不同的destination 地址.
- 这些报文时从 interface Ethernet0/0 发出去的
- 报文的下一跳是 10.200.40.1

- 报文是 ICMP requests (type=8)

在该例中，可以看到IP Input 进程中的 High CPU 是由源自 IP 地址 192.168.40.53 的 ping flood 造成的。

通过这种方式，SYN flood 可以很容易地被发现，因为在 debugging 输出中发现了 SYN flag ：

```
router#show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: disabled
    Monitor logging: disabled
    Buffer logging: level debugging, 145 messages logged
    Trap logging: level informational, 61 message lines logged
Log Buffer (64000 bytes):

*Mar  3 03:43:27.320: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.204
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.324: ICMP type=8, code=0
*Mar  3 03:43:27.324: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.205
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.324: ICMP type=8, code=0
*Mar  3 03:43:27.328: IP: s=192.168.40.53 (Ethernet0/1), d=144.254.2.206
   (Ethernet0/0), g=10.200.40.1, len 100, forward
*Mar  3 03:43:27.328: ICMP type=8, code=0
...
```