

ASR 5000系列排除GTPC和GTPU以及相關路徑故障

目錄

[簡介](#)

[背景](#)

[相關命令](#)

[路徑故障檢測](#)

[路徑故障陷阱和解釋的各種示例](#)

[相關思科支援社群討論](#)

簡介

本文包含了對控制平面(EGTPC)/GTPC(第三代合作夥伴專案技術規範 [29.274](#))、GPRS隧道協定使用者平面(GTPv1-U)(3GPP TS [29.281](#))的瞭解和故障排除，以及與本文相關的最多的恢復過程([3GPP TS 23.007](#))問題，這些問題圍繞計數器、重啟等概念展開路徑故障檢測和回應請求/響應。其中包括相關命令的清單和輸出中要查詢的內容，以及相關可配置檔案的說明。這當然是一個有許多問題的領域。請注意，排除特定呼叫控制問題本身是一個非常廣泛的關注領域，但本文僅提到如何捕獲呼叫控制的統計資訊 — 除此之外，未涵蓋其他任何問題（即呼叫建立問題）。實際上，本文重點介紹排除故障和驗證對等體連線，如果沒有對等體連線，將無法進行呼叫控制。雖然是為操作員編寫的，但設計人員/實施人員也可以從這裡的內容中受益。

示例全部使用來自實際運營商網路的假（已更改）IP地址。

背景

GTP連線的每一端(S5(服務閘道(SGW) — 封包資料網路閘道(PGW))、S11(多媒體管理實體(MME)-SGW)、S4(服務GPRS服務節點(SGSN)-Gateway GPRS服務節點(GGSN))均維持一個重新啟動計數器，用於在始發節點重新啟動時向對等節點發出警報，以便對等節點採取適當的動作。雖然一個節點可以對其所連線的每個對等體使用一個唯一計數器，但設計趨向於對所有對等體使用相同的SAME計數器，因為當節點重新啟動時，需要為所有對等體遞增計數器，因此在對等體基礎上跟蹤單獨的值不會帶來額外的好處，因為不管怎樣，它們都需要增加一個。實際上，對所有對等體使用相同的重新啟動計數器有助於對大型網路進行故障排除，在大型網路中，可以檢查所有對等體，以確保收到的重新啟動計數器對於這些對等體正在通訊的節點具有相同的（單個）值。

重新啟動計數器CAN可以包括在所有型別的控制消息中，而重新啟動計數器MUST必須包括在所有響應請求和響應中。一旦在節點上發生重啟計數器改變，通過傳送到各個對等體的下一控制消息通知各個對等體。根據規範，當重啟發生時，接收方可以選擇暫時保持所擁有的呼叫一段時間，或者丟棄這些呼叫，後者通常是所配置的行為。

在使用回應請求的情況下，該方法是按配置的間隔傳送消息，如果在配置的重試次數（每次在配置的超時後）後未收到響應，則認為路徑已關閉，並採取適當的操作。配置允許可選地傳送回波，這同樣適用於對等體，其方法不一定相同，尤其是當對等體可能來自不同的供應商和/或不同的運營商。事實上，就回應響應而言，節點只負責響應回應請求 — 無論它是否收到回應，以及什麼頻率不重要 — 它只需為每個請求傳送回應響應，並在重新啟動計數器收到更改時採取行動。當然，如前所述

，如果IT同時傳送回應請求並且沒有收到及時響應，它也需要針對該場景採取行動。
注意，路徑故障檢測也可以通過GTP-U（使用者平面）連線通過Echo完成，但只能通過超時完成，因為GTP-U消息中沒有包括重新啟動計數器，與前面討論的GTP-C的重新啟動計數器一樣。
GTP-U使用的路徑可能與控制消息使用的路徑可能相同，也可能不同。如果它是同一路徑，則GTP-U檢測可能沒有那麼有價值，因為GTP-C檢測應該足夠了。
總之，以下是執行路徑故障檢測的方法：

- 重新啟動計數器更改（通過控制消息Req/Rsp或Echo）
- GTPC Echo請求超時
- GTPU回應請求超時
- 控制請求消息超時（通常未配置）

相關命令

若要疑難排解任何型別的路徑問題，請熟悉以下命令。以下是各節相關詳細情況的說明。

- show [egtpc-service | gtpu-service]所有
- show egtpc peers [egtp-service <service>]
- egtpc test echo gtp-version {1 | 2} src-address <service address> peer-address <peer address>
- show egtpc statistics [(sgw-address) | pgw-address | mme-address]<address>] [demux-only] [debug-only] [verbose] | path-failure-reasons]
- show gtpu statistics [peer-address <peer>]
- show gtpc statistics [ggsn-service <service>] [smgr-instance <instance>]
- show demux-mgr statistics <egtpinmgr | egtpegmgr | gtpumgr>所有(SGW-egress | GGSN/PGW, /SGW ingress | GTPU)
- EGTPCPathFail[Clear]、EGTUPPathFail[Clear] SNMP陷阱
- show session disconnect-reasons - gtpc-path-failure, gtpu-path-failure, path-failure

用於計算與GTP相關的訂閱者的其他命令：

show sub [summary] ggsn-service <service>

show sub [summary] gtpu-service <service>

show egtpc sessions [egtp-service <service>]

用於檢查所有EGTP服務的狀態和配置的基本命令是「show egtpc-service」。大部分輸出應該是已通過配置瞭解的，而且狀態應為STARTED。收集到的唯一資訊是「重新啟動計數器」，在其他任何地方都找不到。相關命令「show gtpu-service all」返回與EGTPC服務關聯的GTPU服務的可配置項。

「show egtpc peers」在識別所有對等體時非常有用，包括狀態、重新啟動計數器和當前/最大訂戶計數。

- 請注意，此命令同時涵蓋SGW和SGSN對等體，而過去曾有一個單獨的命令用於SGSN/GGSN。
- 由於此命令包括所有對等體型別，因此可以通過服務名稱過濾對等體型別（即使關鍵字egtpc-service看似也採用GGSN服務名稱）。
- 如果沒有具有對等體的訂閱者，則狀態為Inactive和GTPC Echo disabled。在該狀態下，對於EGTPC，可能存在一些報告Restart計數器的對等體，而大多數可以是0。當對等體正在主動傳送ASR5K應響應的回應請求時，會發生這種情況。重新啟動計數器可以通過運行EGTPC

test命令（下面討論）來確定，儘管在非活動狀態下，計數器不會在「show egtpc peers」中更新，即使它被接收了。同時，對於GGSN，永遠都不會忘記重新啟動計數器。

```
[local]PGW-GGSN> show egtpc peers egtp-service GGSN1
```

Service ID	Peer Address	Restart Counter	No. of restarts	Current sessions	Max sessions
IDNKG 6	203.0.113.69	165	0	0	1
IDNKG 6	203.0.113.70	25	0	0	1
IDNKG 6	203.0.113.71	165	0	0	1
IDNKG 6	203.0.113.72	25	0	0	1
IDNKG 6	203.0.113.73	53	0	0	1
IDNKG 6	203.0.113.74	53	0	0	1
IDNKG 6	192.0.2.9	46	0	0	2
IDNKG 6	192.0.2.10	44	0	0	1
IDNKG 6	192.0.2.11	47	0	0	2
IDNKG 6	192.0.2.12	44	0	0	2
IDNKG 6	192.0.2.13	54	0	0	1
IDNKG 6	192.0.2.14	47	0	0	1
IDNKG 6	209.165.200.225	35	1	0	3
IDNKG 6	209.165.200.226	35	3	0	3
IDNK 6	198.51.100.204	156	0	0	1
IDNK 6	198.51.100.205	158	0	0	1
IDNK 6	198.51.100.211	54	0	0	1
IDNK 6	198.51.100.212	54	0	0	1
IDNK 6	198.51.100.213	56	0	0	1
IDNK 6	198.51.100.214	56	0	0	1
IDNK 6	198.51.100.215	54	0	0	1

請注意，下面幾個對等點的計數遠高於其他對等點。它們用於物理上位於MME發出最多呼叫的PGW的同一站點或姐妹站點中的SGW。

```
[local]PGW> show egtpc peers
Monday March 30 20:10:26 UTC 2015
```

```
+----Status: (I) - Inactive (A) - Active
|+---GTPC Echo: (D) - Disabled (E) - Enabled
||+--Restart Counter Sent: (S) - Sent (N) - Not Sent
|||+--Peer Restart Counter: (K) - Known (U) - Unknown
||||+--Type of Node: (S) - SGW (P) - PGW
|||| (M) - MME (G) - SGSN
|||| (L) - LGW (E) - ePDG
|||| (C) - CGW
|||| (U) - Unknown
```

Service ID	Peer Address	Restart Counter	No. of restarts	Current sessions	Max sessions
IDNUS 5	198.51.100.4	0	0	0	1

IDNKS	5	198.51.100.33	75	0	0	1
IDNUS	5	198.51.100.35	0	0	0	2
IDNUS	5	198.51.100.65	0	0	0	1
IDNUS	5	198.51.100.67	0	0	0	2
IDNUS	5	198.51.100.68	0	0	0	2
IDNUS	5	198.51.100.98	0	0	0	2
IDNUS	5	198.51.100.100	0	0	0	1
IDNUS	5	198.51.100.129	0	0	0	2
IDNUS	5	198.51.100.130	0	0	0	2
IDNUS	5	198.51.100.161	0	0	0	2
IDNUS	5	198.51.100.194	0	0	0	2
IDNUS	5	198.51.100.196	0	0	0	1
IDNUS	5	198.51.100.225	0	0	0	5
IDNUS	5	198.51.100.226	0	0	0	2
IDNUS	5	198.51.100.228	0	0	0	1
IDNUS	5	198.51.100.3	0	0	0	2
IDNUS	5	198.51.100.36	0	0	0	1
IDNUS	5	198.51.100.64	0	0	0	4
IDNUS	5	198.51.100.66	0	0	0	2
IDNUS	5	198.51.100.130	0	0	0	2
IDNKS	5	198.51.100.24	2	0	0	2
AESKS	5	198.51.100.56	3	0	1	5
IDNKS	5	198.51.100.99	18	0	0	2
AESKS	5	198.51.100.100	7	0	1	3
IDNKS	5	198.51.100.131	46	0	0	2
IDNKS	5	198.51.100.132	18	0	0	2
IDNKS	5	198.51.100.163	41	0	0	3
IDNKS	5	198.51.100.164	6	0	0	2
AESKS	5	198.51.100.184	11	0	1	2
IDNKS	5	198.51.100.216	7	0	0	2
AESKS	5	203.0.113.24	5	0	3	8
IDNKS	5	203.0.113.56	3	1	0	13
IDNKS	5	203.0.113.88	2	0	0	18
IDNKS	5	203.0.113.120	6	0	0	3
IDNKS	5	203.0.113.152	3	0	0	4
AESKS	5	203.0.113.184	2	0	1	9
IDNKS	5	203.0.113.216	4	0	0	2
IDNKS	5	203.0.113.248	10	0	0	1
IDNUS	5	203.0.113.4	0	0	0	2
IDNUS	5	203.0.113.33	0	0	0	4
IDNUS	5	203.0.113.65	0	0	0	2
IDNKS	5	203.0.113.121	5	0	0	3
IDNKS	5	203.0.113.153	6	0	0	5
IDNUS	5	203.0.113.162	0	0	0	3
IDNUS	5	203.0.113.32	0	0	0	5
AESKS	5	203.0.113.35	5	1	6	37
IDNUS	5	203.0.113.64	0	0	0	1
IDNUS	5	203.0.113.66	0	0	0	4
IDNUS	5	203.0.113.97	0	0	0	1
AESKS	5	203.0.113.98	4	0	1	4
IDNUS	5	203.0.113.161	0	0	0	4
IDNUS	5	203.0.113.193	0	0	0	10
IDNUS	5	203.0.113.228	0	0	0	12
IDNUS	5	203.0.113.233	0	0	0	4
AESKS	5	192.0.2.4	9	0	3	10
AESKS	5	192.0.2.24	6	0	10	19
AESKS	5	192.0.2.32	8	0	83	116
AESKS	5	192.0.2.33	55	0	14	21
AESKS	5	192.0.2.65	18	0	1	2747
AESKS	5	192.0.2.66	4	0	3	1639
AESKS	5	192.0.2.97	44	39	33	1826
AESKS	5	192.0.2.98	5	1	6	2075

IDNKS	5	192.0.2.195	8	0	0	7
AESKS	5	192.0.2.3	46	0	3230	5346
AESKS	5	192.0.2.5	12	0	84	202
AESKS	5	192.0.2.7	2	0	31	443
AESKS	5	192.0.2.24	3	0	116	117
AESKS	5	192.0.2.36	15	0	4194	6889
AESKS	5	192.0.2.44	7	0	112	331
AESKS	5	192.0.2.68	34	0	6104	9798
AESKS	5	192.0.2.69	7	0	2259	5027
AESKS	5	192.0.2.88	17	0	3899	8880
AESKS	5	192.0.2.164	16	0	833131	1298070
AESKS	5	192.0.2.184	3	0	1437053	2153317
AESKS	5	192.0.2.193	21	0	11364	26577
AESKS	5	192.0.2.195	5	0	50	162
AESKS	5	192.0.2.226	21	0	1950	3931
AESKS	5	192.0.2.227	6	0	4348	9332

請注意，在機箱間服務備援(ICSR)備用節點上運行此命令只與活動對等體相關，在這種情況下，編號應關閉，例如在以下對中：

PGW> show egtpc peers ... 20:51:52(ACTIVE)					PGW-ICSR> show egtpc peers ... 20:51:51 (備用)				
	對等	RC	會話數	馬克斯·塞斯	對等	RC	會話數	馬克斯·塞斯	
AESKS	192.0.2.5	2	31	443	ADSKS	192.0.2.5	2	31	443
AESKS	192.0.2.24	3	117	123	ADNKS	192.0.2.24	3	117	123
AESKS	192.0.2.36	15	4223	6889	ADSKS	192.0.2.36	15	4223	6889
AESKS	192.0.2.44	7	119	331	ADSKS	192.0.2.44	7	119	331
AESKS	192.0.2.68	34	6174	9798	ADSKS	192.0.2.68	34	6171	9797
AESKS	192.0.2.69	7	2311	5027	ADSKS	192.0.2.69	7	2311	5027
AESKS	192.0.2.88	17	3914	8880	ADSKS	192.0.2.88	17	3913	8874
AESKS	192.0.2.164	16	833705	1298070	ADSKS	192.0.2.164	16	832037	1294412
AESKS	192.0.2.184	3	1438418	2153317	ADSKS	192.0.2.184	3	1435461	2147280
AESKS	192.0.2.193	21	11447	26577	ADSKS	192.0.2.193	21	11443	26568
AESKS	192.0.2.195	5	48	162	ADSKS	192.0.2.195	5	48	162
AESKS	192.0.2.226	21	1962	3931	ADSKS	192.0.2.226	21	1962	3931
AESKS	192.0.2.227	6	4465	9332	ADSKS	192.0.2.227	6	4466	9332

「egtpc test echo」用於檢查特定對等體是否可訪問，並且必須在定義服務的上下文中運行。

- 將gtp版本1用於GGSN-SGSN連線
- 使用ping命令不是有效的測試，但是如果成功，則知道存在一定程度的可達性。
- 將顯示對等重新啟動計數器(Recovery)。
- 運行此命令會增加「show egtpc statistics」中的Tx echo請求/Rx Echo響應計數器
- 在Recovery關鍵字之後報告對等體的重新啟動計數器 (下面示例中的11)
- 警告：雖然在正常情況下不應發生這種情況，如果對對等體執行了測試回聲，並且該回聲由於

任何原因而超時（如果所有回聲都在工作，則不應該超時），則所有呼叫都將斷開（即使從測試開始工作到一切正常）。

```
[Ingress]PGW> egtpc test echo gtp-version 2 src-address 209.165.201.1 peer-address 198.51.100.35
Monday November 03 15:08:12 UTC 2014
EGTPC test echo
-----
Peer: 198.51.100.35 Tx/Rx: 1/1 RTT(ms): 1 (COMPLETE) Recovery: 11 (0x0B)
```

「show egtpc statistics」用於EGTPC(v2)，並報告許多EGTPC控制消息（請求/響應、Tx/Rx）的成功/失敗，包括建立會話、修改承載、刪除會話、下行鏈路資料通知、釋放訪問承載、建立承載、更新承載、修改承載、掛起等，以及關聯的響應。它包括回顯請求/響應Tx/Rx的所有路徑管理計數器，並且根據計時器，可以預測這些計時器的增長，並在未按預期增長時用於故障排除。

以下示例顯示從兩端啟動和響應的響應和響應。請注意，SGW已配置為每分鐘傳送一次，而PGW配置為每三分鐘傳送一次（兩端無需具有相同的設定），因此每三分鐘資料包數是原來的兩倍（6:16:33、6:19:33）。請記住，儘管在這個例子中，兩端都處於同一個「循環」中，但它們不一定非要彼此獨立運行。重新啟動計數器也已在列中捕獲。請注意，SGW的回應回應回應中的Restart計數器11與以上對PGW的測試回應的回應中的值相匹配（測試是在與資料包捕獲不同的時間完成的，但針對的是同一對等對）。

No.	Time	Source	Destination	Restart Counter	Info
1	06:15:33	SGW	PGW	11	Echo Request
2	06:15:33	PGW	SGW	18	Echo Response
3	06:16:33	SGW	PGW	11	Echo Request
4	06:16:33	PGW	SGW	18	Echo Response
5	06:16:33	PGW	SGW	18	Echo Request
6	06:16:33	SGW	PGW	11	Echo Response
7	06:17:33	SGW	PGW	11	Echo Request
8	06:17:33	PGW	SGW	18	Echo Response
9	06:18:33	SGW	PGW	11	Echo Request
10	06:18:33	PGW	SGW	18	Echo Response
11	06:19:33	SGW	PGW	11	Echo Request
12	06:19:33	PGW	SGW	18	Echo Response
13	06:19:33	PGW	SGW	18	Echo Request
14	06:19:33	SGW	PGW	11	Echo Response
15	06:20:33	SGW	PGW	11	Echo Request
16	06:20:33	PGW	SGW	18	Echo Response
17	06:21:33	SGW	PGW	11	Echo Request
18	06:21:33	PGW	SGW	18	Echo Response

demux-only選項可捕獲所有路徑管理計數器（回應請求/響應）以及從未到達sessmgr的控制消息（通常是Create Session Request/Response）。駐留在Demux卡上的Demux process etpinmgr負責處理所有回應要求響應和所有呼叫控制消息的初始處理，然後再將其分發到適當的sessmgr中（對於新呼叫是隨機的，特別是對於錨定在特定sessmgr上的現有呼叫）。sessmgr-only選項相反，只計算由sessmgr處理的消息，但不包括路徑管理計數器。不指定任何限定詞是計算所有值的預設值。

「show gtpc statistics」是SGSN/GGSN服務(GTPC v1)（CPC、更新PDP環境等）的類似命令。

「show gtpu statistics」報告與EGTPC服務關聯的GTP-U服務處理的使用者平面流量的統計資訊。一個有趣的計數器是Error Indication Tx/Rx，當接收節點沒有使用者記錄時，將傳送該記錄，該使用者記錄應該與所討論資料包的終端終端識別符號(TEID)相關聯，由於多種原因（批次統計資料方案/變數是GTPU1/err-ind-tx），該記錄可能會發生。如果尚未通知MME/SGW不再具有到訂戶的繫結，則可能需要在MME/SGW上手動清除呼叫（或等待呼叫超時）以完全解決此類孤立呼叫問題。在對特定對等體進行故障排除時，上述stat命令的對等體特定版本是必需的，這種情況通常存在。

路徑故障檢測

EGTPCPathFail陷阱是瞭解路徑故障發生時間的關鍵。在已經假設的一些基本值中，它會報告已丟棄的呼叫數、舊重啟計數器和新重啟計數器以及失敗的原因。此外，會話斷開原因 gtpc-path-failure將遞增。

EGTPCPathFailClear陷阱將指示重新啟動計數器以及路徑清除的原因。最新的StarOS v17的措辭最好是「明確原因」，而不是「失敗原因」（它被清除了，而不是失敗了！），以避免混淆。以下是從 show egtpc stat path-failure-reasons報告的路徑故障計數

```
[local]PGW> show egtpc statistics path-failure-reasons
Reasons for path failure at EGTPC:
Echo Request restart counter change:          3
Echo Response restart counter change:          0   echo-rsp-restart-counter-change
No Echo Response received:                    0   no-response-from-peer
Control message restart counter change at demux: 23 Create Session Req Restart Counter changed or
Control message restart counter change at sessmgr: 0   cpc-restart-counter-change or upc-restart-counter-change
Total path failures detected:                  26
```

附註：

- sessmgr上的控制消息重新啟動計數器可以是任何型別的控制消息（此處未列出所有控制消息）
- 更改後，通過下一個建立PDP上下文請求從SGSN更改的重新啟動計數器(cpc-restart-counter-change)
 - 在更改之後通過下一更新PDP上下文請求從SGSN更改的重啟計數器（這將是從另一個SGSN的切換）(upc-restart-counter-change)
 - Path clear failure reason sessions-add是一個欺騙性的術語 — 它不是故障，而是成功，表示已進入新會話以建立連線
 - 如果PathFail和PathClear陷阱的重新啟動計數器和會話計數報告的值似乎沒有排隊，請開啟思科的票證以獲得進一步說明。

重要的是要瞭解EGTPC/GTPC和較小程度的EGTPU/GTPU（由於使用者平面故障檢測並非總是被配置）如何實現路徑故障檢測，以監控GGSN或PGW與各種對等SGSN和SGW之間的連線。通常情況下，服務提供商網路中的連線相當穩定，在這種情況下路徑故障較為少見。另一方面，經驗表明，連線到GGSN的漫遊SGSN之間的GTPC路徑可能不是那麼穩定，因此，如果連線不佳，就會導致發生相當頻繁的路徑故障。

以下配置涵蓋了可用於檢測路徑故障的控制和回應消息的計時器和傳輸頻率。同樣的概念適用於GGSN、SGW、MME和EGTP服務：

ggsn服務GGSN1

retransmission-timeout 20	對於GTP控制消息（與回應消息相對）
最大重新傳輸5	對於GTP控制和回應消息 — 在觸發路徑故障檢測之前重新傳輸的次數
echo-interval 120	每2分鐘（120秒）傳送一次回應。
echo-retransmission-timeout 5	對於Echos，如果沒有響應，則每5秒重新傳輸一次
path-failure detection-policy gtp echo	通過gtp echo可以檢測路徑故障(與不將其用作檢測路徑故障的方法（但仍在傳送回聲）相反)。

關聯gtpu服務 GGSN1_gtpu	將命名GTPU服務GGSN1_gtpu與此GGSN服務關聯
bind ipv4-address 209.165.201.2	以下示例配置中的使用者和控制平面繫結IP地址相同，但可能未必與其他服務提供商實現配置相同。

```
[local]GGSN> show ggsn-service name GGSN1
```

```
Service name:          GGSN1
Context:              XGWin
Bind:                Done
Local IPv4 Address:    209.165.201.2      Local IP Port: 2123
Retransmission Timeout: 20 (secs)
Max Retransmissions:   5
Restart Counter:       41
Echo Interval:         120 (secs)
GTPC Echo Mode                : Default
GTPC Echo Retransmission Timeout : 5
Guard Interval:             100 (secs)
Path Failure Detection on gtp msgs: Echo
```

附註：

- 給定時間0是回應要求週期開始時（每2分鐘），如果總共30秒（6次嘗試）後沒有響應，則路徑將失敗(0 + 5秒+ (5 * 5秒))
- Process egtpinmgr（自v14以來不再支援gtpcmgr）（運行在demux卡上）負責處理回應資料包
- "show gtp statistics","Path Management Messages"部分包含兩個方向的控制路徑的回應請求/響應的統計資訊：

```
[Ingress]GGSN> show gtpc statistics
Monday October 27 23:02:07 UTC 2014
```

Path Management Messages:

```
Echo Request RX:          169951248    Echo Response TX:          169951248
Echo Request TX:          40838432     Echo Response RX:          40784820
```

另一種路徑檢測方法是在使用者平面上而不是控制平面上，這由以下配置控制。

```
gtpu-service GGSN1_gtpu
no echo-interval
max-retransmissions 4
echo-retransmission-timeout 5
path-failure detection-policy gtp echo
no path-failure clear-trap gtp echo
bind ipv4-address 209.165.201.2
exit
```

附註：

- 「no echo-interval」禁止向使用者平面傳送回應請求，因此CLI「path-failure detection-policy gtp echo」沒有相關性 — NO Path failure detection is done on GTPU
- 雖然GGSN仍會響應來自SGSN的回應請求，
- "show gtpu stat"和"show demux-mgr stat gtpumgr"捕獲回顯統計資訊

```
[local]PGW-GGSN> show gtpu statistics
Wednesday October 29 23:43:22 UTC 2014
Path Management Messages:
Echo Request Rx:          84690303  Echo Response Rx:          0
Echo Request Tx:          0  Echo Response Tx:      84690043
```

```
[local]PGW-GGSN> show demux-mgr statistics gtpumgr
Wednesday October 29 23:42:04 UTC 2014
Echo stats:
Echo Req Rcvd:            84690303
Echo Rsp Sent:            84690043
```

路徑故障陷阱和解釋的各種示例

通常，當存在重啟時，對等重啟計數器將增加1。在這種情況下，PGW中的錯誤使其將重新啟動計數器重寫為另一個對等體的RC值（第一個陷阱），因此當下一個資料包到達連線時，它將其視為重新啟動（第二個陷阱），認為RC已更改。接著，路徑會在任何連續封包（第三陷阱）上清除。此處的要點不是瞭解特定錯誤，而是嘗試並瞭解無法解釋的任何計數器更改。

```
Wed Jan 14 05:22:07 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Wed Jan 14 05:22:07 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Wed Jan 14 05:22:07 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service EGTP1, i
```

此示例來自對等體SGSN通過建立PDP環境請求將其RC從66更改為67的GGSN:

```
Mon Oct 27 11:48:54 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service GGSN1, i
Mon Oct 27 11:48:54 2014 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service GGSN1, i
```

此範例來自對等體ePDG已透過建立作業階段要求將其RC從69變更為70的PGW:

```
Fri Jun 12 05:21:54 2015 Internal trap notification 1112 (EGTPCPathFail) context PGWin, service EGTP1, i
Fri Jun 12 05:21:55 2015 Internal trap notification 1113 (EGTPCPathFailClear) context PGWin, service EGTP1, i
```

在本示例中，來自所有連線到同一SGW的各個PGW節點的輸出顯示，在彼此的短時間幀內

，SGW停止從各PGW上每個EGTPCPathFail陷阱的PGW訪問。請注意，所有節點都顯示與前面討論的相同舊重啟計數器，節點通常對所有對等體使用相同的計數器。如果沒有響應，則顯示的新重新啟動計數器為9，但新計數器可能會列印為未定義，因為連線從陷阱觸發時起已關閉，因此使用重新啟動計數器實際上沒有意義。

```
Mon Dec 22 09:07:29 2014 Internal trap notification 1112 (EGTPCPathFail) context PGWin, service EGTP1, i
Mon Dec 22 09:08:54 2014 Internal trap notification 1112 (EGTPCPathFail) context PGWin, service EGTP1, i
Mon Dec 22 09:09:00 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Mon Dec 22 09:07:42 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Mon Dec 22 09:08:06 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Mon Dec 22 09:07:34 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Mon Dec 22 09:08:20 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Mon Dec 22 09:06:37 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Mon Dec 22 09:07:29 2014 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Mon Dec 22 09:07:01 2014 Internal trap notification 1112 (EGTPCPathFail) context PGWin, service EGTP1, i
```

以下示例顯示一個路徑故障和與前一個示例相同的新重新啟動計數器，但是半小時之後顯示對等體重新啟動並且計數器從15增加到16時增加了一個的次數。隨著時間的推移，該對等體的計數增加：

```
<#root>

Wed Jul 22 05:07:08 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Wed Jul 22 05:38:00 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service EGTP1, i

16

, peer session count 1, clear reason sessions-update

[local]PGW> show egtpc peers
Wednesday July 22 10:04:30 UTC 2015

||||| Service      Restart-----+ No. of
||||| ID           Counter   | restarts
||||| |            |      | |
vvvvv v            Peer Address v  v  Current      Max
-----
AESKS  5           198.51.100.162

16

1      1157      2186

[XGWin]PGW> egtpc test echo gtp-version 2 src-address 209.165.201.13 peer-address 198.51.100.162
Wednesday July 22 10:52:41 UTC 2015
EGTPC test echo
-----
Peer: 198.51.100.162          Tx/Rx: 1/1 RTT(ms): 51 (COMPLETE) Recovery:
```

(0x10)

此示例顯示由於CPC重啟導致GGSN的 **3個特定SGSN**對等點出現重複路徑故障。下面的輸出只顯示其中一個對等路由器209.165.200.225，但它們都存在相同的問題。重複運行「show egtpc peers」和「show gtpc stat sgsn-address」也會顯示交替值。「show gtpc stat」可確認與回應請求和響應相關的連線沒有問題，在這種情況下，此連線僅由GGSN啟動（如前面討論的，可接受）。這裡需要指出的一個側邊是，通常可以收集到多個資料，所有這些資料都印證了同一個結論 — 人們可以選擇以最小的努力收集最確定的資料。

<#root>

```
show snmp trap history verbose | grep -E "209.165.201.31|209.165.200.225|209.165.200.246"
```

Mon Jun 22 22:20:51 2015 Internal trap notification 1112 (EGTPathFail) context XGWin, service GGSN1, i

132

```
, peer session count 3, failure reason
```

cpc-restart-counter-change

Mon Jun 22 22:20:51 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service GGS

```
Mon Jun 22 22:20:55 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service GGSN1, i
```

129

```
, peer session count 3, failure reason cpc-restart-counter-change
```

Mon Jun 22 22:20:55 2015 Internal trap notification 1113 (EGTPathFailClear) context XGWin, service GGS

Mon Jun 22 22:20:58 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service GGSN1, i

132

```
, peer session count 3, failure reason cpc-restart-counter-change
```

Mon Jun 22 22:20:58 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service GGS

```
Mon Jun 22 22:21:02 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service GGSN1, i
```

129

```
, peer session count 2, failure reason cpc-restart-counter-change
```

Mon Jun 22 22:21:02 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service GGS

Mon Jun 22 22:21:03 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service GGSN1, i

132

```
, peer session count 4, failure reason echo-rsp-restart-counter-change
```

```
Mon Jun 22 22:21:03 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service GGS
```

Mon Jun 22 22:21:06 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service GGSN1, i

129

```
, peer session count 2, failure reason cpc-restart-counter-change
```

Mon Jun 22 22:21:06 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service GGS

```
[XGWin]GGSN> show eqtpc peers | grep 209.165.200.225
```

	Service		Restart	No. of		
	ID		Counter	restarts		
					Current	Max
vvvvv	v	Peer Address			sessions	sessions
			v	v	v	

Monday June 22 23:09:26 UTC 2015

AESKG 6 209.165.200.225

129

86339

2 57

[XGWin]GGSN> show egtpc peers | grep 209.165.200.225

Monday June 22 23:09:28 UTC 2015

AESKG 6 209.165.200.225

129

86341 3 57

[XGWin]GGSN> show egtpc peers | grep 209.165.200.225

Monday June 22 23:09:29 UTC 2015

AESKG 6 209.165.200.225

132

86342 7 57

[XGWin]GGSN> show egtpc peers | grep 209.165.200.225

Monday June 22 23:09:30 UTC 2015

AESKG 6 209.165.200.225

132

86344 9 57

[XGWin]GGSN> show egtpc peers | grep 209.165.200.225

Monday June 22 23:09:31 UTC 2015

AESKG 6 209.165.200.225

129

86345 11 57

[XGWin]GGSN> show egtpc peers | grep 209.165.200.225

Monday June 22 23:09:32 UTC 2015

AESKG 6 209.165.200.225

129

86345

7 57

[XGWin]GGSN> show gtpc statistics sgsn-address 209.165.200.225

Monday June 22 23:07:36 UTC 2015

SGSN Address: 209.165.200.225 Status: Active

Total Restarts:

86287

Restart Counter:

129

```

Session Stats:
  Total Current:          2    S6b Assume Positive:          0
  ...

Path Management Messages:
  Echo Request RX:        0    Echo Response TX:          0
  Echo Request TX:

```

22919

Echo Response RX:

22917

檢視子網中的其他對等體，它們始終顯示RC = 132，而斷開的對等體在129和132之間交替顯示（如圖所示），這表示這些SGSN對等體在斷開的情況下可能會傳遞錯誤的RC:

<#root>

		Restart Counter					
AESKG	6	209.165.200.225	129				
85115							
8	57	<==					
IDNKG	6	209.165.200.226	132	21	0	3	
AESKG	6	209.165.200.227	132	60	47	92	
AESKG	6	209.165.200.228	132	53	24	59	
AESKG	6	209.165.200.229	132	69	92	126	
AESKG	6	209.165.200.232	132	66	21	30	
AESKG	6	209.165.200.236	132	43	8	21	
AESKG	6	209.165.200.238	132	49	18	32	
AESKG	6	209.165.200.239	132	39	1	6	
AENKG	6	209.165.200.240	132	17	1	5	
AESKG	6	209.165.200.241	132	67	6	43	
AESKG	6	209.165.200.242	132	56	22	29	
AENKG	6	209.165.200.243	132	40	2	9	
AENKG	6	209.165.200.246	129				
17713							
3	24	<==					
AESKG	6	209.165.200.247	132	64	29	45	
AESKG	6	209.165.200.250	132	62	42	56	

回想前文所述，解決此類問題的另一種方法是檢查其他GGSN（或LTE的PGW），以確定它們是否與那些看似有問題的SGSN對等體存在同樣的問題，在這種情況下，它們也存在同樣的問題。對於GGSN3，第一次運行該命令時，兩個對等體的值都顯示為132，但是另一個運行則顯示其中一個對等體的129。多次運行（此處未顯示是為了簡潔起見）將顯示這些對等點的所有GGSN在兩個RC之間隨時間跳變：

```

[local]GGSN1> show egtpc peers | grep -E "209.165.200.246|209.165.200.225"
Tuesday June 23 11:37:23 UTC 2015
AENKG    7          209.165.200.246 129 20147          1          29
AESKG    7          209.165.200.225 129 92853          2          51

[local]GGSN2> show egtpc peers | grep -E "209.165.200.246|209.165.200.225"

```

```

Tuesday June 23 11:37:23 UTC 2015
AENKG 6 209.165.200.246 129 20450 4 17
AESKG 6 209.165.200.225 129 94021 3 54

[local]GGSN3> show egtpc peers | grep -E "209.165.200.246|209.165.200.225"
Tuesday June 23 11:37:26 UTC 2015
AENKG 6 209.165.200.246 132 20211 1 39
AESKG 6 209.165.200.225 132 91996 3 59

[local]GGSN4> show egtpc peers | grep -E "209.165.200.246|209.165.200.225"
Tuesday June 23 11:37:25 UTC 2015
AENKG 6 209.165.200.246 129 20309 1 24
AESKG 6 209.165.200.225 132 92694 9 57

[local]GGSN3> show egtpc peers | grep -E "209.165.200.246|209.165.200.225"
Tuesday June 23 11:44:35 UTC 2015
AENKG 6 209.165.200.246 129 20242 3 39
AESKG 6 209.165.200.225 132 92224 1 59

```

最後，要確認問題在於SGSN更改RC，SGSN對等體監控用戶選單選項顯示更改RC。當它發生更改時，所有呼叫都將斷開，**path-failure**將作為斷開原因。封包擷取截圖也會顯示以下內容：

<#root>

```

[XGWin]GGSN> mon sub
Monday June 22 23:34:22 UTC 2015

```

y) By SGSN IP Address

IP Address: [209.165.200.225]

```

INBOUND>>>> 23:35:32:688 Eventid:47000(3)
GTPC Rx PDU, from 209.165.200.225:2123 to 209.165.202.129:2123 (176)
TEID: 0x00000000, Message type: GTP_CREATE_PDP_CONTEXT_REQ_MSG (0x10)
Sequence Number:: 0x1DB8 (7608)
...

```

IMSI: 300420078559902

Recovery: 0x81 (129)

...

```

***CONTROL*** 23:36:00:363 Eventid:10285
Disconnect Reason: path-failure

```

```

INBOUND>>>> 23:36:48:414 Eventid:47000(3)
GTPC Rx PDU, from 209.165.200.225:2123 to 209.165.202.129:2123 (176)
TEID: 0x00000000, Message type: GTP_CREATE_PDP_CONTEXT_REQ_MSG (0x10)
Sequence Number:: 0x3E2D (15917)
...

```

IMSI: 300420125984926

Recovery: 0x84 (132

)
...

INBOUND>>>> 23:37:28:337 Eventid:47000(3)
 GTPC Rx PDU, from 209.165.200.225:2123 to 209.165.202.129:2123 (176)
 TEID: 0x00000000, Message type: GTP_CREATE_PDP_CONTEXT_REQ_MSG (0x10)
 Sequence Number:: 0x3517 (13591)
 ...
 IMSI: 300420094205377

Recovery: 0x84 (132)

...

Monday June 22 2015

INBOUND>>>> 23:37:40:559 Eventid:47000(3)
 GTPC Rx PDU, from 209.165.200.225:2123 to 209.165.202.129:2123 (176)
 TEID: 0x00000000, Message type: GTP_CREATE_PDP_CONTEXT_REQ_MSG (0x10)
 Sequence Number:: 0x4E47 (20039)
 ...
 IMSI: 300420194755472

Recovery: 0x81 (129)

...

CONTROL 23:37:40:755 Eventid:10285
 Disconnect Reason: path-failure

No.	Time	Source	Destination	Recovery	Info
42	17:50:03	SGSN	GGSN	129	Create PDP context request
63	17:50:03	SGSN	GGSN	129	Create PDP context request
68	17:50:04	SGSN	GGSN	129	Create PDP context request
69	17:50:04	SGSN	GGSN	129	Create PDP context request
70	17:50:04	SGSN	GGSN	132	Create PDP context request
94	17:50:06	SGSN	GGSN	129	Create PDP context request
157	17:50:11	SGSN	GGSN	132	Create PDP context request
164	17:50:12	SGSN	GGSN	132	Create PDP context request
166	17:50:12	SGSN	GGSN	129	Create PDP context request
167	17:50:12	SGSN	GGSN	132	Create PDP context request
169	17:50:13	SGSN	GGSN	132	Create PDP context request
171	17:50:13	SGSN	GGSN	132	Create PDP context request
195	17:50:14	SGSN	GGSN	129	Create PDP context request
209	17:50:16	SGSN	GGSN	132	Create PDP context request
249	17:50:17	SGSN	GGSN	129	Create PDP context request
397	17:50:18	SGSN	GGSN	132	Create PDP context request
398	17:50:18	SGSN	GGSN	129	Create PDP context request
436	17:50:20	SGSN	GGSN	129	Create PDP context request
490	17:50:22	SGSN	GGSN	129	Create PDP context request
495	17:50:23	SGSN	GGSN	129	Create PDP context request
688	17:50:26	SGSN	GGSN	132	Create PDP context request
756	17:50:27	SGSN	GGSN	132	Create PDP context request
772	17:50:30	SGSN	GGSN	132	Create PDP context request
774	17:50:31	SGSN	GGSN	132	Create PDP context request
779	17:50:31	SGSN	GGSN	132	Create PDP context request
855	17:50:32	SGSN	GGSN	129	Create PDP context request

此範例顯示一個路徑故障，其中問題實際上出在路徑中，而不是PGW或SGW。因此，重新建立路徑時RC會保持不變。在第二個對等示例中，clear是建立回應要求的結果：

Thu May 28 20:47:47 2015 Internal trap notification 1112 (EGTPCPathFail) context PGWin, service EGTP1, i
Thu May 28 20:49:36 2015 Internal trap notification 1113 (EGTPCPathFailClear) context PGWin, service EGT
Thu May 28 20:49:06 2015 Internal trap notification 1112 (EGTPCPathFail) context PGWin, service EGTP1, i
Thu May 28 20:50:49 2015 Internal trap notification 1113 (EGTPCPathFailClear) context PGWin, service EGT

此範例顯示另一個異常情況，即在舊重新啟動計數器重設為2後，若路徑故障清除，它不知何故再次以0結束。這原來是一個錯誤。同樣，這裡的教訓是，如果出現奇怪的情況，就應該予以報告。

Fri Mar 27 03:13:36 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Fri Mar 27 03:13:37 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service EGT
Sat Mar 28 02:20:45 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Sat Mar 28 02:20:47 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service EGT
Sat Mar 28 17:14:06 2015 Internal trap notification 1112 (EGTPCPathFail) context XGWin, service EGTP1, i
Sat Mar 28 17:14:06 2015 Internal trap notification 1113 (EGTPCPathFailClear) context XGWin, service EGT

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。