ASR 5000系列排除GTPC和GTPU及相关路径故障

目录

简介

背景

相关命令

路径故障检测

路径故障陷阱的各种示例和说明

相关的思科支持社区讨论

简介

本文涉及对控制平面(EGTPC)/GTPC（第三代合作伙伴项目技术规范 29.274）、GPRS隧道协议用户平面(GTPv1-U)(3GPP TS 29.281)的演进通用分组无线业务隧道协议(EGTPC)/GTPC（第三代合作伙伴项目技术规范 29.274）的了解和故障排除，以及与本文最相关的恢复过程 (3GPP TS 23.007)问题，这些问题围绕诸如计数器、重启等概念路径故障检测和回应请求/响应。其中包括相关命令列表和要在输出中查找的内容，以及相关可配置文件的说明。这当然是一个有许多疑问的领域。请注意，排除特定呼叫控制问题本身就是一个非常广泛的兴趣领域，但本文仅提到如何捕获呼叫控制的统计信息 — 除此之外的所有问题（即呼叫建立问题）均未涉及。实际上，本文重点介绍排除故障和检验对等体连接，如果没有对等体连接，将无法进行呼叫控制。 虽然是为操作人员编写的，但设计人员/实施人员也可以从此处的内容中受益。
示例全部使用来自真实运营商网络的假（已更改）IP地址。

背景

GTP连接的每一端(S5(服务网关(SGW) — 数据包数据网络网关(PGW))、S11(多媒体管理实体(MME)- SGW)、S4(服务GPRS服务节点(SGSN) — 网关GPRS服务节点(GGSN)之一)维护一个重启计数器，该计数器设计用于在源节点重新启动时向对等体发出警报，以便对等体可以采取适当的操作。虽然一个节点可以对其所交谈的每个对等体使用唯一的计数器，但设计通常是对所有对等体使用相同的计数器，因为当节点重新启动时，所有对等体的计数器需要递增，因此在对等体基础上跟踪单独的值不会带来额外的好处，因为它们无论如何都需要递增1。实际上，对所有对等体使用相同的重新启动计数器有助于排除大型网络的故障，在此网络中，可以检查所有对等体，以确保收到的重新启动计数器对于这些对等体正在通信的节点具有相同的（单个）值。
重新启动计数器可以包含在所有类型的控制消息中，而重新启动计数器必须包含在所有回应请求和响应中。一旦在节点上发生重启计数器改变，通过发送到各个对等体的下一控制消息通知各个对等体。根据规范，当重启发生时，接收方可以选择暂时保持其已进行的呼叫一段时间，或者丢弃它们，后者通常是已配置的行为。
在使用回应请求的情况下，方法是在配置的间隔内发送消息，如果在配置的重试次数（每次在配置的超时后）后未收到响应，则认为路径已关闭并采取适当的操作。 配置允许可选地发送回声，这同样适用于对等体，对等体的方法不一定相同，特别是当对等体可能来自不同的供应商和/或不同的运营商。事实上，就Echo响应而言，节点仅有义务响应Echo请求 — 无论其是否接收echo以及什么频

率不重要 — 它只需为每个请求发送echo响应，并在重新启动计数器收到更改时采取行动。当然，如前所述，如果IT部门也发送回应请求且未收到及时响应，则还需要针对该场景采取行动。
请注意，路径故障检测也可以通过GTP-U（用户平面）连接通过Echo完成，但只能通过timeout完成，因为GTP-U消息中不包含重启计数器，与前面讨论的GTP-C的重启计数器相 同。GTP-U使用的路径可能与控制消息使用的路径相同，也可能不同。如果它是同一路径，则GTP-U检测可能没有那么有价值，因为GTP-C检测应该足够了。
总之，以下是进行路径故障检测的方法：

- 重新启动计数器更改（通过控制消息Req/Rsp或Echo）
- GTPC回应请求超时
- GTPU回应请求超时
- 控制请求消息超时（通常未配置）

相关命令

要排除任何类型的路径问题，请熟悉以下命令。以下是对每项服务的相关详细信息的说明。

- 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出口 | GGSN/PGW，/SGW入口 | GTPU)
- EGTPCPathFail[Clear]、EGTPUPathFail[Clear] SNMP陷阱
- show session disconnect-reasons - gtpc-path-failure， gtpu-path-failure， path-failure

用于计算与GTP相关的用户的其它命令：
show sub [summary] ggsn-service <service>
show sub [summary] gtpu-service <service>
show egtpc sessions [egtp-service <service>]
用于检查所有EGTP服务的状态和配置的基本命 令是“show egtpc-service”。 大部分输出应该是已通过配置了解的，并且状态应为STARTED。收集到的唯一信息是“Restart Counter”，在其他任何地方都找不到。相关命 令“show gtpu-service all”返回与EGTPC服务关联的GTPU服务的可配置项。
“show egtpc peers”在识别所有对等体时非常有用，包括状态、重新启动计数器和当前/最大用户计数。

- 请注意，此命令同时涵盖SGW和SGSN对等体，而过去曾有一个用于SGSN/GGSN的单独命令。
- 由于此命令包括所有对等体类型，因此可以通过服务名称过滤对等体类型（即使关键字egtpc-service看似也采用GGSN服务名称）。
- 如果没有具有对等体的用户，则状态为Inactive和GTPC Echo disabled。在该状态下，对于EGTPC，可能存在为其报告重启计数器的对等体，而大多数可以为0。 当对等体主动发送ASR5K应响应的回应请求时，会出现这种情况。重新启动计数器可以通过运行EGTPC测试命

令（下面将讨论）来确定，但在非活动状态下，计数器不会在“show egtpc peers”中更新，即使计数器已收到。 同时，对于GGSN，永远都不会忘记重新启动计数器。

```
[local]PGW-GGSN> show egtpc peers egtp-service GGSN1
|||||  Service                        Restart--+   No. of
|||||  ID                             Counter  |  restarts
|||||   |                                      |     |     Current   Max
vvvvv   v              Peer Address            v     v     sessions 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
```

请注意，下方的两个对等体比其他对等体重要得多。这些用于物理上位于MME发出最多呼叫的PGW的同一站点或姐妹站点中的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                       Restart--------+  No. of
|||||  ID                            Counter        |  restarts
|||||   |                                           |   |   Current       Max
vvvvv   v              Peer Address                 v   v   sessions    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
```

请注意，在机箱间服务冗余(ICSR)备用节点上运行此命令仅与活动对等体相关，在这种情况下，编号应接近，例如以下对中的编号：

PGW> show egtpc peers ... 20:51:52（活动）						PGW-ICSR> show egtpc peers ... 20:51:51（备用）			
	对等体	RC	会话数量	最大Sess	对等体	RC	会话数量	最大Sess	
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”用于检查特定对等体是否可访问，并且必须在定义服务的上下文中运行。

- 将gtp版本1用于GGSN-SGSN连接
- 使用ping命令不是有效的测试，但如果成功，则表明存在一定程度的连通性。
- 系统将显示对等重启计数器（恢复）。
- 运行此命令将增加“show egtpc statistics”中的Tx echo request / Rx Echo响应计数器
- 在Recovery关键字后报告对等体的重新启动计数器（以下示例中为11）
- 警告：虽然这种情况在正常情况下不应发生，但如果在对等体上执行了测试回声，并且该回声由于任何原因而超时（如果所有回声都在工作，则不应超时），则所有呼叫都会掉线（即使直

到执行测试时一切工作正常）。

```
[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”用于EGTPC(v2)，并报告许多EGTPC控制消息（请求/响应、Tx/Rx）的成功/失败，包括创建会话、修改承载、删除会话、下行链路数据通知、释放接入承载、创建承载、更新承载、修改承载、挂起等，以及关联的响应。它包括Echo Request/Response Tx/Rx的所有路径管理计数器，根据计时器，可以预测这些计时器的增长，并在不按预期增长时用于排除故障。
以下示例显示从两端发起和响应的回应和响应。请注意，SGW已配置为每分钟发送一次，而PGW已配置为每三分钟发送一次（两端无需具有相同的设置），因此每三分钟数据包数是原来的两倍(6:16:33、6:19:33)。 请记住，尽管在本例中，两端都处于同一个“循环”中，但它们不一定非得是而且可以彼此独立运行。 重新启动计数器也在列中捕获。请注意，来自SGW的回应响应的重新启动计数器11与来自以上的PGW的测试回应的响应值相匹配（测试是在与数据包捕获不同的时段完成的，但针对的是同一对等对）。

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选项可捕获所有路径管理计数器（echo请求/响应）以及从未到达sessmgr的控制消息（通常为Create Session Request/Response）。驻留在Demux卡上的Demux process etpinmgr负责处理所有回声请求/响应和所有呼叫控制消息的初始处理，然后再将其分发到适当的sessmgr（新呼叫的随机处理，特别是锚定在特定sessmgr上的现有呼叫）。 相反， sessmgr-only选项仅计算由sessmgr处理的消息，但不包括路径管理计数器。不指定任何限定词是计算所有值的默认值。
“show gtpc statistics”是SGSN/GGSN服务(GTPC v1)（CPC、更新PDP环境等）的类似命令
“show gtpu statistics”报告与EGTPC服务关联的GTP-U服务处理的用户平面流量的统计信息。 一个有趣的计数器是Error Indication Tx/Rx，当接收节点没有用户记录时，将发送该错误指示Tx/Rx，该用户记录应该与数据包的终端终端标识符(TEID)相关联，这可能由于多种原因而发生（批量统计数据方案/变量为GTPU1/ err-ind-tx）。 如果MME/SGW未获通知PGW不再具有到用户的绑定，则可能需要在MME/SGW上手动清除呼叫（或等待呼叫超时）以完全解决此类孤立呼叫问题。
在对特定对等体进行故障排除时，上述stat命令的对等体特定版本是必需的，这种情况通常存在。

路径故障检测

EGTPCPathFail陷阱是了解路径故障发生时间的关键。在已经假设的一些基本值中，它会报告已丢弃的呼叫数、旧重启计数器和新重启计数器以及故障原因。此外，会话断开原因 gtpc-path-failure将递增。
EGTPCPathFailClear陷阱将指示重新启动计数器以及路径清除的原因。最新的StarOS v17的措辞最好是“明确原因”，而不是“失败原因”（清除而不是失败！），以避免混淆。
以下是从 show egtpc stat path-failure-reasons报告的路径故障计数

```
[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-cou
 Total path failures detected:                       26
```

注意：

- sessmgr处的控制消息重新启动计数器可以是任何类型的控制消息（并非此处列出所有控制消息）
- 更改后，通过下一个创建PDP环境请求从SGSN更改重新启动计数器(cpc-restart-counter-change)
  - 更改后通过下一个更新PDP环境请求从SGSN更改的重新启动计数器（这将是从另一个SGSN的切换）(upc-restart-counter-change)
  - Path clear failure reason sessions-add是一个欺骗性术语 — 它不是故障，而是成功，表示已进入新会话以建立连接
  - 如果PathFail和PathClear陷阱的重新启动计数器和会话计数的报告值似乎未排队，请打开思科的票证进行进一步说明。

重要的是要了解EGTPC/GTPC和较低程度的EGTPU/GTPU（因为用户平面故障检测并非始终配置）路径故障检测如何实施，以监控GGSN或PGW与各种对等SGSN和SGW之间的连接。通常，服务提供商网络中的连接相当稳定，在这种情况下路径故障并不常见。另一方面，经验表明，连接到GGSN的漫游SGSN中的GTPC路径可能不是很稳定，因此，如果连接不佳，将导致路径故障相当频繁地发生。
以下配置包括可用于检测路径故障的控制和回应消息的计时器和传输频率。同样的概念适用于GGSN、SGW、MME和EGTP服务：

ggsn-service GGSN1

retransmission-timeout 20	对于GTP控制消息（与回应消息相对）
最大重新传输数5	对于GTP控制和回应消息 — 在触发路径故障检测之前重新传输的次数
echo-interval 120	每2分钟（120秒）发送一次响应。
echo — 重新传输超时5	对于Echos，如果没有响应，则每5秒重新传输一次
path-failure detection-policy gtp echo	通过gtp echo可以检测路径故障(而不是将其用作检测路径故障的方法（但仍在发送回声）)。
associate gtpu-service	将命名GTPU服务GGSN1_gtpu与此GGSN服务关联

GGSN1_gtpu	
bind ipv4-address 209.165.201.2	以下示例配置中的用户和控制平面绑定IP地址相 同，但可能未必在其他服务提供商实施中如此配置。

```
[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
```

注意：

- 假定响应请求周期开始的时间0（每2分钟），如果总时间为30秒（6次尝试）后没有响应，则路径会失败(0 + 5秒+（5 * 5秒））
- 在解复用器卡上运行的Process egtpinmgr（自v14以来已弃用gtpcmgr）负责处理回应数据包
- “show gtp statistics”，“Path Management Messages”部分在两个方向上都有控制路径的回应请求/响应的统计信息：

```
[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
```

另一种路径检测方法是在用户平面而不是控制平面上，这由以下配置控制。

```
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
```

注意：

- “no echo-interval”禁止向用户平面发送回应请求，因此CLI“path-failure detection-policy gtp echo”没有相关性 — NO Path failure detection在GTPU上完成
- 虽然GGSN仍然响应来自SGSN的回应请求，
- "show gtpu stat"和"show demux-mgr stat gtpumgr"捕获响应统计信息

```
[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
```

路径故障陷阱的各种示例和说明

通常，当存在重启时，对等重启计数器将增加1。在本例中，PGW中的漏洞导致它使用另一个对等体（第一个陷阱）的RC值覆盖重新启动计数器，因此当下一个数据包通过连接到达时，它将其视为重新启动（第二个陷阱），认为RC已更改。然后，路径会清除任何连续数据包（第三个陷阱）。这里的要点不是为了知道某个特定的Bug，而是尝试了解无法解释的任何计数器更改。

```
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 EGT
```

此示例来自对等体SGSN通过创建PDP环境请求将其RC从66更改为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 GGS
```

此示例来自对等体ePDG通过创建会话请求将其RC从69更改为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 EGT
```

在本示例 中，连接到同一SGW的各个PGW节点的输出显示，SGW在彼此的短时间帧内停止在各个PGW上通过EGTPCPathFail陷阱从PGW到达。请注意，所有节点均显示与前面讨论过的相同旧重

启计数器，即节点通常对所有对等体使用相同的计数器。如果没有响应，则显示的新重新启动计数器为9，但新计数器可能最好打印为未定义值，因为从触发陷阱时起，连接已关闭，因此使用重新启动计数器实际上没有意义。

```
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
```

以下示例显示一个路径故障和与前一个示例相同的新重新启动计数器，但半小时后，显示对等体再次启动且计数器已从15增加到16时增加了一个。随着时间的推移，该对等体的计数增加：

<#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 EGT

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
|||||   |                               |    |   Current       Max
vvvvv   v             Peer Address     v    v   sessions    sessions
-----  --- ---------------------------------------- --- --- -----------
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:

16

 (0x10)
```

此示例显示由于CPC重启而导致GGSN 的3个特定SGSN对等体出现重复路径故障。下面的输出只显示一个对等体209.165.200.225，但它们都有相同的问题。重复运行“show egtpc peers”和“show gtpc stat sgsn-address”也会显示交替值。“show gtpc stat”可确认与回应请求和响应相关的连接没有问题，在这种情况下，只有GGSN发起连接（如前面讨论的一样）。 这里需要强调的一点是，通常可以收集到多个数据，所有这些数据都印证了同一个结论 — 你可以选择收集最确凿的数据，同时尽量少花精力。

<#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 (EGTPCPathFail) 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 (EGTPCPathFailClear) 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 egtpc 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
```

查看子网中的其他对等体，它们始终显示RC = 132，而中断的对等体在129和132之间交替（如图所示），这表示这些SGSN对等体在中断的情况下可能会传递错误的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
```

前面讲过，解决此类问题的另一种方法是检查其他GGSN（或LTE的PGW），看看它们是否与那些看似有问题的SGSN对等体存在同样的问题，在本例中它们也存在同样的问题。对于GGSN3，第一次运行该命令时，两个对等体的值都显示为132，而另一个运行则显示其中一个对等体的129。多次运行（此处未显示是为了简洁起见）将显示这些对等体的两个RC之间的所有GGSN随时间跳转：

```
[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
```

最后，要确认问题是SGSN更改RC，SGSN对等体的monitor subscriber菜 单选项显示更改RC。当它发生更改时，所有呼叫都会断开， path-failure作为断开原因。数据包捕获的屏幕截图还显示了以下内容：

<#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

此示例显示路径故障，其中问题实际上在路径中，而不是PGW或SGW。因此，重新建立路径时，RC保持不变。在第二个对等示例中，clear是建立回应请求的结果：

```
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
```

此示例显示另一种异常，在旧重启计数器重置为2后，当路径故障清除时，它会反复以0结束。这原来是一个漏洞。同样，这里的教训是，如果出现奇怪的情况，就应该予以报告。

```
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
```

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。