

Serie ASR 5000: Fehlerbehebung bei GTPC und GTPU sowie damit verbundenen Pfadausfällen

Inhalt

[Einleitung](#)

[Hintergrund](#)

[Relevante Befehle](#)

[Erkennung von Pfadausfällen](#)

[Verschiedene Beispiele für Pfadfehler-Traps und Erklärungen](#)

[Verwandte Diskussionen in der Cisco Support Community](#)

Einleitung

Dieser Artikel behandelt das Verständnis und die Fehlerbehebung von Evolved General Packet Radio Service Tunneling Protocol for Control Plane (EGTPC) / GTPC (3rd Generation Partnership Project Technical Specification [29.274](#)), GPRS Tunneling Protocol User Plane (GTPv1-U) (3GPP TS [29.281](#)) und die dafür relevantesten Artikel, Restoration Procedures ([3GPP TS 23.007](#)) Probleme , die sich um Konzepte wie Neustartzähler, Pfadausfallerkennung und Echo-Anfragen/Antworten drehen. Es enthält eine Liste relevanter Befehle und Informationen dazu, was in der Ausgabe gesucht werden sollte, sowie eine Erläuterung der zugehörigen konfigurierbaren Optionen. Dies ist sicherlich ein Bereich, für den viele Fragen aufgeworfen werden. Beachten Sie, dass die Behebung bestimmter Probleme bei der Anrufsteuerung an sich ein sehr umfangreicher Bereich ist. In diesem Artikel wird jedoch nur erwähnt, wie Statistiken für die Anrufsteuerung erfasst werden können. Alle darüber hinausgehenden Fragen (z. B. Probleme bei der Anrufeinrichtung) werden nicht behandelt. Dieser Artikel befasst sich hauptsächlich mit der Fehlerbehebung und dem Überprüfen der Peer-Verbindungen. Ohne diese Verbindungen gäbe es keine Anrufsteuerung. Während für Betreiber geschrieben, können Designer / Implementierer auch von den Inhalten hier profitieren.

In allen Beispielen werden gefälschte (geänderte) IP-Adressen aus dem Netzwerk eines realen Betreibers verwendet.

Hintergrund

Jede Seite einer GTP-Verbindung (S5 (Serving Gateway (SGW) - Packet Data Network Gateway (PGW)), S11 (Multimedia Management Entity (MME) - SGW), S4 (Serving GPRS Service Node (SGSN) - Gateway GPRS Service Node) (GGSN)) einen Neustartzähler verwaltet, der den Peer benachrichtigen soll, wenn der ursprüngliche Knoten neu gestartet wurde, damit dieser die entsprechenden Maßnahmen ergreifen kann. Während ein Knoten für jeden Peer, mit dem er kommuniziert, einen eindeutigen Zähler verwenden kann, wird im Design tendenziell derselbe Zähler für alle Peers verwendet, da beim Neustart des Knotens der Zähler für alle Peers inkrementiert werden muss. Die Verfolgung eines separaten Werts auf Peer-Basis hat daher keinen zusätzlichen Vorteil, da sie alle ohnehin um einen erhöhen müssen. In der Tat ist die

Verwendung des gleichen Neustartzählers für alle Peers hilfreich bei der Fehlerbehebung in einem großen Netzwerk, in dem alle Peers überprüft werden können, um sicherzustellen, dass die empfangenen Neustartzähler den gleichen (einzelnen) Wert für den Knoten haben, mit dem diese Peers sprechen.

Der Neustart-Zähler KANN in allen Arten von Kontrollmeldungen enthalten sein, während der Neustart-Zähler in allen Echo-Anfragen und -Antworten enthalten sein MUSS. Sobald auf einem Knoten ein Neustart-Zählerwechsel erfolgt, werden die verschiedenen Peers über die nächsten an die jeweiligen Peers gesendeten Steuernachrichten darüber informiert. Bei einem Neustart hat die empfangende Seite gemäß Spezifikation die Möglichkeit, die Anrufe vorübergehend für eine kurze Zeit aufrechtzuerhalten oder zu verwerfen, letzteres ist in der Regel das konfigurierte Verhalten. Bei der Verwendung von Echo-Anfragen besteht die Methode darin, die Nachrichten in einem konfigurierten Intervall zu senden. Wenn nach einer konfigurierten Anzahl von Wiederholungen (jeweils nach einem konfigurierten Timeout) keine Antwort empfangen wird, gilt der Pfad als ausgefallen und die entsprechende Aktion wird ausgeführt. Die Konfiguration ermöglicht das optionale Senden von Echos. Dies gilt auch für den Peer, dessen Ansatz nicht unbedingt derselbe sein muss, insbesondere dann, wenn der Peer von einem anderen Anbieter und/oder einem anderen Netzbetreiber stammt. In Bezug auf Echo-Antworten ist ein Knoten nur verpflichtet, auf eine Echo-Anfrage zu antworten - unabhängig davon, ob er Echos empfängt oder nicht und mit welcher Frequenz er nicht signifikant ist - er muss lediglich eine Echo-Antwort für jede Anfrage senden und Maßnahmen ergreifen, wenn sich der empfangene Neustartzähler ändert. Natürlich, wie bereits erwähnt, WENN es auch Echo-Anfragen sendet und nicht rechtzeitig Antworten erhält, muss es auch für dieses Szenario Maßnahmen ergreifen.

Beachten Sie, dass die Erkennung von Pfadausfällen auch über die GTP-U-Verbindung (Benutzerebene) über Echo erfolgen kann, jedoch NUR über Timeout, da in GTP-U-Nachrichten kein Neustart-Zähler enthalten ist, wie er für GTP-C bereits beschrieben wurde. GTP-U verwendet möglicherweise den gleichen Pfad wie Kontrollnachrichten. Wenn es sich um denselben Pfad handelt, ist die GTP-U-Erkennung möglicherweise weniger nützlich, da die GTP-C-Erkennung ausreichen sollte.

Zusammenfassend lassen sich folgende Vorgehensweisen aufzeigen, mit denen Pfadausfälle erkannt werden können:

- Zählerwechsel neu starten (über Steuerung msg Req/Rsp oder Echo)
- Timeout für GTPC-Echoanforderung
- Zeitüberschreitung bei GTPU-Echoanforderung
- Timeout für Kontrollanforderungsnachricht (normalerweise nicht konfiguriert)

Relevante Befehle

Machen Sie sich mit den folgenden Befehlen vertraut, um Probleme mit dem Pfad zu beheben. Im Folgenden werden die jeweiligen Details erläutert.

- `show [egtpc-service] | gtpu-service] alle`
- `show egtpc peers [egtp-service <Service>]`
- `egtpc test echo gtp-Version {1 | 2} src-Adresse <Dienstadresse> Peer-Adresse <Peer-Adresse>`
- `show egtpc statistics [(sgw-address) | PGW-Adresse | me-address) <Adresse>] [nur demux] [nur debug] [verbose] | Gründe für Pfadfehler]`

- show gtpu statistics [peer-address <peer>]
- show gtpc statistics [ggsn-service <service>] [smgr-instance <instance>]
- show demux-mgr statistics <egtpinmgr | egtpmgr | gtpmgr> alle (SGW-Ausgang) | GGSN/PGW-/SGW-Eingang | GTPU)
- EGTPCPathFail[Clear], EGTPUPathFail[Clear] SNMP-Traps
- show session disconnect-reason - gtpc-path-failure, gtpu-path-failure, path-failure

Weitere Befehle zum Zählen von Abonnenten in Bezug auf GTP:

show sub [summary] ggsn-service <service>

show sub [summary] gtpu-service <service>

show egtpc sessions [egtp-service <Service>]

Der einfache Befehl zum Überprüfen des Status und der Konfiguration aller EGTP-Dienste lautet "show egtpc-service". Die meisten Ausgaben sollten so sein, wie sie in der Konfiguration bereits bekannt sind, und der Status sollte GESTARTET sein. Die einzigartigen Informationen, die gesammelt wurden, sind der "Neustart-Zähler", der nirgendwo anders gefunden werden kann. Der zugehörige Befehl "show gtpu-service all" gibt konfigurierbare Werte für die GTPU-Dienste zurück, die den EGTPC-Diensten zugeordnet sind.

"show egtpc peers" ist sehr nützlich, um alle Peers zu identifizieren, einschließlich Status, Neustartzähler und aktuelle/maximale Abonnentenanzahl.

- Beachten Sie, dass dieser Befehl sowohl SGW- als auch SGSN-Peers abdeckt, während es in der Vergangenheit einen separaten Befehl für SGSN/GGSN gab.
- Da dieser Befehl alle Peer-Typen umfasst, kann der Peer-Typ über den Dienstnamen gefiltert werden (obwohl das Schlüsselwort "egtpc-service" trügerisch auch einen GGSN-Dienstnamen akzeptiert).
- Wenn es keine Abonnenten mit dem Peer gibt, lautet der Status Inaktiv und GTPC-Echo deaktiviert. In diesem Zustand kann es für EGTPC einige Peers geben, für die der Neustart-Zähler gemeldet wird, während die meisten 0 sein können. Dies tritt in Situationen auf, in denen der Peer aktiv Echo-Anfragen sendet, auf die ASR5K reagieren sollte. Der Zähler für den Neustart kann durch Ausführen des EGTPC-Testbefehls bestimmt werden (siehe nächstes Thema). Im inaktiven Zustand wird der Zähler jedoch nicht in "show egtpc peers" aktualisiert, obwohl er empfangen wurde. Für GGSN wird der Neustart-Zähler derweil nie vergessen.

[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

Beachten Sie unten einige Peers, deren Anzahl wesentlich höher ist als die der anderen Peers. Diese sind für SGWs vorgesehen, die sich physisch am selben Standort befinden, oder für den Schwesterstandort des PGW, an den die MME die meisten Anrufe weiterleitet.

```
[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

Beachten Sie, dass die Ausführung dieses Befehls auf einem Inter-Chassis Service Redundancy (ICSR) Standby-Knoten nur für aktive Peers relevant ist. In diesem Fall sollten die Nummern nah sein, z. B. im folgenden Paar:

PGW> egtpc-Peers anzeigen ... 20:51:52 (AKTIV)					PGW-ICSR> egtpc-Peers anzeigen ... 20:51:51 (STANDBY)				
	Peer	RC	Anzahl Sitzungen	Max Sess		Peer	RC	Anzahl Sitzungen	Max 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" wird verwendet, um einen bestimmten Peer auf Erreichbarkeit zu überprüfen und muss in dem Kontext ausgeführt werden, in dem der Service definiert ist.

- gtp-Version 1 für GGSN-SGSN-Verbindungen verwenden
- Die Verwendung des Befehls ping ist kein gültiger Test. Wenn der Test erfolgreich ist, gibt es jedoch eine gewisse Erreichbarkeit.
- Der Zähler für den Peer-Neustart (Wiederherstellung) wird angezeigt.
- Wenn Sie diesen Befehl ausführen, werden die Zähler für Tx-Echoanfrage/Rx-Echoantwort in "show egtpc statistics" erhöht.
- Der Neustartzähler des Peers wird nach dem Wiederherstellungs-Schlüsselwort gemeldet (im Beispiel unten 11).
- Warnung: Auch wenn dies unter normalen Umständen nicht der Fall sein sollte, würden alle Anrufe abgebrochen, wenn ein Peer-Echo durchgeführt wird und es aus irgendeinem Grund zu einer Zeitüberschreitung kommt (wenn alles funktioniert, sollte es nicht zu einer Zeitüberschreitung kommen) (obwohl die Dinge bis zum Testzeitpunkt einwandfrei funktionierten).

```
[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" ist für EGTPC (v2) und berichtet über den Erfolg/Misserfolg vieler EGTPC-Kontrollmeldungen (Request/Response, Tx/Rx), einschließlich Create Session, Modify Bearer, Delete Session, Downlink Data Notify, Release Access Bearer, Create Bearer, Update Bearer,

Modify Bearer, Suspend, etc., und den zugehörigen Antworten. Er enthält alle Pfadmanagement-Zähler für Echo Request/Response Tx/Rx. Basierend auf den Timern kann die Zunahme dieser Timer vorhergesagt und zur Fehlerbehebung verwendet werden, wenn sie nicht erwartungsgemäß erhöht wird.

Das folgende Beispiel zeigt Echos und Reaktionen, die von beiden Seiten initiiert und beantwortet werden. Beachten Sie, dass das SGW so konfiguriert wurde, dass es einmal pro Minute sendet, während das PGW so konfiguriert wurde, dass es einmal alle drei Minuten sendet (beide Seiten müssen nicht die gleiche Einstellung haben), sodass alle drei Minuten doppelt so viele Pakete gesendet werden (6:16:33, 6:19:33). Beachten Sie, dass beide Seiten zwar in diesem Beispiel im gleichen "Zyklus" sind, aber nicht unabhängig voneinander sein müssen und laufen können. Die Neustartzähler wurden ebenfalls in einer Spalte erfasst. Beachten Sie, dass der Neustartzähler von 11 aus der Echo-Antwort des SGW mit dem Wert aus seiner Antwort auf das PGW-Testecho von oben übereinstimmt (der Test wurde zu einem anderen Zeitpunkt durchgeführt als die Paketerfassung, jedoch für dasselbe Peer-Paar).

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

Die demux-only-Option erfasst alle Pfadmanagement-Zähler (Echo-Anfragen/Antworten) sowie Steuermeldungen, die nie einen Sessmgr erreichen, in der Regel Create Session Request/Response (Session-Anfrage/Antwort erstellen). Der auf der Demux-Karte befindliche Demux-Prozess etpinmgr ist für die Verarbeitung aller Echo-Anfragen/-Antworten und die Erstbehandlung aller Anrufsteuerungsmeldungen vor deren Verteilung an die entsprechenden Sessmgr (zufällig für neue Anrufe und speziell für bestehende Anrufe, die auf einem bestimmten Sessmgr verankert sind) verantwortlich. Die Option sessmgr-only zählt umgekehrt nur Nachrichten, die von sessmgrs verarbeitet werden. Diese Option enthält KEINE Pfadmanagement-Zähler. Wenn Sie keinen der beiden Kennzeichner angeben, werden standardmäßig alle Werte gezählt.

"show gtpc statistics" ist der analoge Befehl für SGSN/GGSN-Dienste (GTPC v1) (CPC, Update PDP Context, etc.)

"show gtpu statistics" meldet Statistiken zum Datenverkehr auf Benutzerebene, der von dem/den GTP-U-Dienst(en) verarbeitet wird/werden, der/die mit dem/den EGTPC-Dienst(en) verknüpft ist/sind. Ein interessanter Zähler ist die Fehleranzeige Tx/Rx, die gesendet wird, wenn der empfangende Knoten keinen Datensatz des Teilnehmers hat, der dem Endpunkt-Identifizierer (TEID) des betreffenden Pakets zugeordnet werden soll, was aus verschiedenen Gründen passieren könnte (Schema/Variable bulkstats ist GTPU1/ err-ind-tx). Wenn das MME/SGW nicht darüber informiert wurde, dass das PGW keine Bindung mehr zum Teilnehmer hat, müssen

Anrufe möglicherweise manuell im MME/SGW gelöscht werden (oder warten, bis das Timeout erreicht ist), um ein solches Problem von verwaisten Anrufen vollständig zu beheben. Die Peer-spezifischen Versionen der oben genannten stat-Befehle sind erforderlich, wenn Sie Fehler bei bestimmten Peers beheben, was häufig der Fall ist.

Erkennung von Pfadausfällen

Der EGTPCPathFail -Trap ist der Schlüssel zum Erkennen eines Pfadfehlers. Unter einigen Grundwerten, die man bereits annehmen würde, sind die Anzahl der abgebrochenen Anrufe, der alte und der neue Zähler für den Neustart und der Grund für den Fehler zu finden. Darüber hinaus wird der Grund für den Verbindungsabbruch "gtpc-path-failure" erhöht.

Das EGTPCPathFailClear-Trap gibt den Neustartzähler zusammen mit dem Grund für das Löschen des Pfads an. Das neueste StarOS v17 ist besser formuliert, um Verwirrung zu vermeiden, statt "Fehlergrund" zu sagen (es wurde gelöscht, nicht fehlgeschlagen!).

Es folgen Pfadausfallzählungen, die aus " show egtpc stat path-failure-reason" gemeldet werden.

```
[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	

Hinweise:

- Der Zähler für den Neustart der Steuernachricht bei sessmgr kann jeder Typ einer Steuernachricht sein (hier nicht alle aufgeführt).
- Neustartzähler vom SGSN über die nächste Create PDP Context Request nach dieser Änderung geändert (cpc-restart-counter-change)
 - Neustartzähler vom SGSN über die nächste Update PDP Context Request nach dieser Änderung geändert (dies wäre eine Übergabe von einem anderen SGSN) (upc-restart-counter-change)
 - Path clear failure reason sessions-add ist ein irreführender Begriff - es ist kein Fehler, sondern ein Erfolg, und es weist darauf hin, dass eine neue Sitzung eingetreten ist, um die Verbindung herzustellen
 - Wenn die für die Neustartzähler und die Sitzungszähler für die PathFail- und PathClear-Traps gemeldeten Werte nicht übereinstimmen, öffnen Sie ein Ticket bei Cisco, um weitere Erläuterungen zu erhalten.

Es ist wichtig zu verstehen, wie EGTPC/GTPC und in geringerem Maße EGTPU/GTPU (da die Erkennung von Ausfällen auf Benutzerebene nicht immer konfiguriert ist) implementiert werden, um die Verbindung zwischen einem GGSN oder PGW und den verschiedenen Peer-SGSN und SGWs zu überwachen. In der Regel sind die Verbindungen innerhalb eines Service-Provider-Netzwerks relativ stabil, sodass es nur selten zu Pfadausfällen kommt. Andererseits hat die Erfahrung gezeigt, dass GTPC-Pfade zwischen Roaming-SGSNs, die mit GGSNs verbunden sind, möglicherweise nicht so stabil sind. Bei schlechten Verbindungen führt dies daher zu häufigen Pfadausfällen.

Die folgende Konfiguration deckt Timer und Übertragungsfrequenzen für Steuerungs- und Echomeldungen ab, die zum Erkennen von Pfadfehlern verwendet werden können. Das gleiche Konzept gilt für GGSN-, SGW-, MME- und EGTP-Services:

ggsn-Service GGSN1

Zeitüberschreitung bei erneuter Übertragung 20	Für GTP-Kontrollnachrichten (im Gegensatz zu Echo-Nachrichten)
Max. Neuübertragungen 5	für BEIDE GTP-Steuerungs- und Echo-Meldungen - Anzahl der erneuten Übertragungen vor dem Auslösen der Pfadausfallerkennung
Echo-Intervall 120	Alle 2 Minuten (120 Sekunden) wird ein Echo gesendet.
Zeitüberschreitung bei der Echorücksendung 5	Falls Echos nicht antwortet, wird die Nachricht alle 5 Sekunden erneut gesendet.
path-failure detection-policy gtp echo	Die Erkennung von Pfadausfällen ist über GTP-Echo möglich (im Gegensatz dazu kann diese Methode nicht zur Erkennung von Pfadausfällen verwendet werden (es werden jedoch weiterhin Echos gesendet)).
Zuordnen - GTPU-Dienst GGSN1_gtpu	Verbindet den benannten GTPU-Dienst GGSN1_gtpu mit diesem GGSN-Dienst
bind ipv4-Adresse 209.165.201.2	<u>Die</u> Binding-IP-Adressen für <u>Benutzer</u> und <u>Kontrollebene</u> sind in den folgenden Beispielkonfigurationen <u>identisch</u> , können jedoch nicht notwendigerweise zusammen mit anderen Service-Provider-Implementierungen als solche konfiguriert werden.

```
[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
```

Hinweise:

- Wenn 0 ist, wenn ein Echo-Anforderungszyklus beginnt (alle 2 Minuten), wenn es nach INSGESAMT 30 Sekunden (6 Versuche) keine Antworten gibt, schlägt der Pfad fehl (0 + 5 Sekunden + (5 * 5 Sekunden))
- Der Prozess "egtpinmgr" (gtpcmgr veraltet seit v14), der auf der Demux-Karte ausgeführt wird, ist für die Verarbeitung der Echo-Pakete verantwortlich.
- "show gtp statistics", Abschnitt "Path Management Messages" enthält Statistiken für die

Echo-Anfragen/-Antworten für den Steuerungspfad in BEIDE Richtungen:

```
[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

Eine weitere Methode zur Pfaderkennung ist auf der Benutzerebene anstelle der Kontrollebene verfügbar. Sie wird durch die folgende Konfiguration geregelt.

```
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
```

Hinweise:

- "no echo-interval" deaktiviert das Senden von Echo-Anfragen für die Benutzerebene. Daher ist CLI "path-failure detection-policy gtp echo" irrelevant - auf GTPU wird KEINE Pfaderkennung durchgeführt.
- GGSN reagiert jedoch weiterhin auf Echoanfragen vom SGSN.
- "show gtpu stat" und "show demux-mgr stat gtpumgr" erfassen Echo-Statistiken

```
[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

Verschiedene Beispiele für Pfadfehler-Traps und Erklärungen

Normalerweise erhöht sich der Zähler für den Peer-Neustart um eins, wenn ein Neustart stattfindet. In diesem Fall führte ein Fehler im PGW dazu, dass der Zähler für den Neustart auf

den Wert des RC eines anderen Peers (erstes Trap) überschrieben wurde. Wenn das nächste Paket über die Verbindung eintraf, wurde es als Neustart (zweites Trap) behandelt, da man davon ausging, dass sich der RC geändert hatte. Der Pfad wird dann bei jedem nachfolgenden Paket gelöscht (dritter Trap). Der Punkt ist hier nicht über einen bestimmten Fehler zu wissen, sondern zu versuchen und sich über alle Zähleränderungen bewusst zu sein, die nicht erklärt werden können.

```
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
```

Dieses Beispiel stammt aus einem GGSN, bei dem der Peer-SGSN seine RC durch eine Create PDP Context Request von 66 auf 67 geändert hat:

```
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
```

Dieses Beispiel stammt von einem PGW, bei dem das Peer-ePDG seine RC durch eine Create Session Request von 69 auf 70 geändert hat:

```
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
```

In diesem Beispiel zeigen die Ausgaben verschiedener PGW-Knoten, die alle mit demselben SGW verbunden sind, an, dass das SGW nicht mehr von den PGWs über einen EGTPCPathFail-Trap auf den verschiedenen PGWs innerhalb eines kurzen Zeitrahmens erreichbar ist. Beachten Sie, dass alle denselben alten Neustartzähler zeigen, wie zuvor erläutert wurde, dass ein Knoten normalerweise denselben Zähler für alle Peers verwendet. Im Falle einer fehlenden Antwort ist der neu angezeigte Neustart-Zähler 9, aber der neue Zähler kann besser als undefiniert gedruckt werden, da die Verbindung zum Zeitpunkt der Auslösung des Traps abgeschaltet ist und ein Neustart-Zähler daher nicht wirklich sinnvoll ist.

```
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

```

Im folgenden Beispiel werden ein Pfadfehler und derselbe Zähler für den neuen Neustart wie im vorherigen Beispiel gezeigt. Eine halbe Stunde später wird jedoch angezeigt, wenn der Peer wieder aktiviert wird und der Zähler um einen Wert von 15 auf 16 erhöht wurde. Im Laufe der Zeit wird die Anzahl für diesen Peer erhöht:

```
<#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
vvvvv	v	Peer Address	v
			Current
			Max
			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)
```

Dieses Beispiel zeigt einen Fehler bei einem wiederholten Pfad aufgrund eines **CPC-Neustarts** für drei spezifische SGSN-Peers zum GGSN. Die folgende Ausgabe zeigt nur einen der Peers 209.165.200.225, aber alle haben das gleiche Problem. Der wiederholte Lauf von "show egtpc peers" und "show gtpc stat sgsn-address" zeigt auch die alternierenden Werte an. "show gtpc stat" bestätigt, dass keine Verbindungsprobleme in Bezug auf Echo-Anfragen und -Antworten bestehen, die in diesem Fall nur vom GGSN initiiert werden (akzeptabel wie weiter oben beschrieben). Ein Seitenleistenpunkt, der hier angesprochen werden muss, ist, dass es oft mehrere Daten gibt, die gesammelt werden können, die alle zum selben Schluss führen - man kann wählen, die Daten zu sammeln, die am aussagekräftigsten sind, mit dem minimalen

Aufwand.

<#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 GGSN1, i

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 GGSN1, i

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 GGSN1, i

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 GGSN1, i

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 GGSN1, i

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 GGSN1, i

```
[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:

Wenn Sie sich andere Peers im Subnetz ansehen, zeigen sie alle ständig einen RC = 132, während die fehlerhaften Peers zwischen 129 und 132 abwechseln, wie gerade gezeigt, was darauf hindeutet, dass diese SGSN-Peers in den fehlerhaften Szenarien möglicherweise den falschen RC passieren:

<#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

Wie bereits erwähnt, besteht ein anderer Ansatz bei Problemen wie diesem darin, andere GGSNs (oder PGWs für LTE) zu überprüfen, ob sie dasselbe Problem mit diesen scheinbar problematischen SGSN-Peers haben, was in diesem Fall auch der Fall ist. Für GGSN3 zeigte der erste Befehl Werte 132 für beide Peers an, während ein anderer Befehl Werte 129 für einen Peer anzeigte. Mehrere Runs (hier aus Gründen der Kürze nicht dargestellt) würden alle GGSNs anzeigen, die für diese Peers im Laufe der Zeit zwischen den beiden RCs hin- und herspringen:

```
[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
```

Schließlich zur Bestätigung des Problems ist der SGSN ändern die RC, überwachen Abonnenten Menüoption durch SGSN Peer zeigt wechselnde RCs. Bei einer solchen Änderung werden alle Anrufe getrennt, wobei als Grund für die Verbindungstrennung **path-failure** angegeben wird. Ein Screenshot einer Paketerfassung zeigt außerdem Folgendes:

```
<#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

Dieses Beispiel zeigt einen Pfadfehler, bei dem das Problem tatsächlich im Pfad lag und nicht beim PGW oder SGW. Dadurch bleibt der RC beim Wiederherstellen des Pfades gleich. Im zweiten Peer-Beispiel ist clear ein Ergebnis der Einrichtung einer Echoanforderung:

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 EGTP1, i

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

Dieses Beispiel zeigt eine weitere Anomalie, bei der der alte Zähler nach dem Zurücksetzen auf 2 wiederholt bei 0 endet, wenn der Pfad nicht freigegeben wird. Das stellte sich als Fehler heraus. Auch hier gilt: Wenn etwas komisch erscheint, sollte es gemeldet werden.

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

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.