

Debug Shell中的Linux TCP转储，用于排除Diameter连接问题

目录

[简介](#)

[背景信息](#)

[对等问题](#)

[数据包捕获收集](#)

[PCAP分析](#)

简介

本文档介绍如何在StarOS调试外壳中使用TCP转储来排除Diameter连接问题。通常会出现问题，请求帮助排除Diameter连接未接通或断开的原因，即使（据说）未发生配置或网络更改。在初始TCP/IP协商级别或在功能交换请求(CER)/功能交换应答(CEA)级别，直径连接可能无法建立。

背景信息

虽然不存在典型的直径对等问题，但它们确实属于以下几类：

- 特定终端和/或协议的所有对等体都关闭。 <====本示例
- 特定端口号的对等体关闭。
- 与特定PSC、DPC或SF卡绑定的对等体已关闭。

通常，TCP端口3868（默认）用于Diameter服务器端，但也可以指定其他端口，并且如果对等配置线路在行尾指定了端口号，则确认该端口与配置中的3868不同。

对等问题

在本例中，终端3gpp-aaa-s6b的对等体通过**show diameter peer full all**向下报告，并且在对等体线路中未指定端口#，因此默认情况下使用端口3868，而Gy的对等体使用不同对等体的3868、3869和3870组合。

show diameter peers all报告所有直径端点的所有已配置对等体。此处我们看到为3gpp-aaa-s6b（已断开）和Gy（工作）配置了6个对等体和关联的配置行，并注意到Gy有一些自定义端口#s：

```
diameter endpoint 3gpp-aaa-s6b

origin realm epc.mnc260.mcc310.3gppnetwork.org
use-proxy
origin host s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org address 10.168.86.144
max-outstanding 64
route-failure threshold 100
route-failure deadtime 600
route-failure recovery-threshold percent 50
dscp af31
peer mp2.daldra01.dra.epc.mnc260.mcc310.3gppnetwork.org realm epc.mnc260.mcc310.3gppnetwork.org
```

```
address 10.160.113.136
peer mp2.elgdra01.dra.epc.mnc260.mcc310.3gppnetwork.org realm epc.mnc260.mcc310.3gppnetwork.org
address 10.160.114.136
peer mp2.nvldra01.dra.epc.mnc260.mcc310.3gppnetwork.org realm epc.mnc260.mcc310.3gppnetwork.org
address 10.160.115.136
peer tsa06.draaro01.dra.epc.mnc260.mcc310.3gppnetwork.org realm
epc.mnc260.mcc310.3gppnetwork.org address 10.162.6.73
peer tsa06.drasyo01.dra.epc.mnc260.mcc310.3gppnetwork.org realm
epc.mnc260.mcc310.3gppnetwork.org address 10.164.57.41
peer tsa06.drawsc01.dra.epc.mnc260.mcc310.3gppnetwork.org realm
epc.mnc260.mcc310.3gppnetwork.org address 10.177.70.201
route-entry peer mp2.daladra01.dra.epc.mnc260.mcc310.3gppnetwork.org
route-entry peer mp2.elgdra01.dra.epc.mnc260.mcc310.3gppnetwork.org
route-entry peer mp2.nvldra01.dra.epc.mnc260.mcc310.3gppnetwork.org
route-entry peer tsa06.draaro01.dra.epc.mnc260.mcc310.3gppnetwork.org
route-entry peer tsa06.drasyo01.dra.epc.mnc260.mcc310.3gppnetwork.org
route-entry peer tsa06.drawsc01.dra.epc.mnc260.mcc310.3gppnetwork.org

#exit
```

```
[local]IEPCF201# show diameter peers all
```

```
Friday December 11 20:27:43 UTC 2020
```

```
Diameter Peer details
```

```
=====
```

```
-----
Context: billing                Endpoint: 3gpp-aaa-s6b
-----
```

```
Peer: mp2.daladra01.dra.epc.mnc260.mc Addr:Port 10.160.113.136:3868
```

```
Peer: mp2.elgdra01.dra.epc.mnc260.mc Addr:Port 10.160.114.136:3868
```

```
Peer: mp2.nvldra01.dra.epc.mnc260.mc Addr:Port 10.160.115.136:3868
```

```
Peer: tsa06.draaro01.dra.epc.mnc260. Addr:Port 10.162.6.73:3868
```

```
Peer: tsa06.drasyo01.dra.epc.mnc260. Addr:Port 10.164.57.41:3868
```

```
Peer: tsa06.drawsc01.dra.epc.mnc260. Addr:Port 10.177.70.201:3868
-----
```

```
diameter endpoint credit-control
```

```
origin realm starent.gy.com
```

```
use-proxy
```

```
origin host iepcf201.gy address 10.168.86.151
```

```
destination-host-avp always
```

```
route-failure threshold 100
```

```
route-failure deadtime 600
```

```

route-failure recovery-threshold percent 50
peer ln24.daladra01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.160.113.136
port 3869
peer ln24.drawsc01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.177.70.201
port 3870
peer tsa05.drachr01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.164.144.88
peer tsa05.draphx01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.198.93.88
peer tsa05.drapol01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.182.16.88
peer tsa06.drachr01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.164.144.89
peer tsa06.draphx01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.198.93.89
peer tsa06.drapol01.dra.epc3.mnc260.mcc310.3gppnetwork.org realm nsn-gy address 10.182.16.89
route-entry peer ln24.drawsc01.dra.epc3.mnc260.mcc310.3gppnetwork.org weight 20
route-entry peer ln24.daladra01.dra.epc3.mnc260.mcc310.3gppnetwork.org
route-entry peer tsa05.drapol01.dra.epc3.mnc260.mcc310.3gppnetwork.org
route-entry peer tsa06.drapol01.dra.epc3.mnc260.mcc310.3gppnetwork.org
route-entry peer tsa05.drachr01.dra.epc3.mnc260.mcc310.3gppnetwork.org weight 5
route-entry peer tsa05.draphx01.dra.epc3.mnc260.mcc310.3gppnetwork.org weight 5
route-entry peer tsa06.drachr01.dra.epc3.mnc260.mcc310.3gppnetwork.org weight 5
route-entry peer tsa06.draphx01.dra.epc3.mnc260.mcc310.3gppnetwork.org weight 5

#exit

```

同样值得注意的是，对于大多数设置，use-proxy configurable指定为在ASR端设置对等，以使用在所有活动卡上运行的diamproxy进程，例如，这是一个vPC-DI，其中卡称为服务功能卡。

```

[local]IEPCF201# show task resources facility diamproxy all
Friday December 11 20:34:37 UTC 2020

```

| cpu facility | task inst | cputime | | memory | | files | | sessions | | | S status | |
|--------------|-----------|---------|-------|--------|--------|--------|------|----------|------|----|----------|------|
| | | used | allc | used | alloc | used | allc | used | allc | | | |
| 3/0 | diamproxy | 5 | 0.12% | 90% | 41.62M | 250.0M | 38 | 2500 | -- | -- | - | good |
| 5/0 | diamproxy | 2 | 0.11% | 90% | 41.63M | 250.0M | 51 | 2500 | -- | -- | - | good |
| 6/0 | diamproxy | 6 | 0.13% | 90% | 41.62M | 250.0M | 35 | 2500 | -- | -- | - | good |
| 7/0 | diamproxy | 3 | 0.12% | 90% | 41.64M | 250.0M | 34 | 2500 | -- | -- | - | good |
| 8/0 | diamproxy | 4 | 0.13% | 90% | 41.65M | 250.0M | 34 | 2500 | -- | -- | - | good |
| 10/0 | diamproxy | 1 | 0.10% | 90% | 41.64M | 250.0M | 49 | 2500 | -- | -- | - | good |
| Total | | 6 | 0.71% | | 249.8M | | 241 | | | | | 0 |

```

[local]IEPCF201#

```

此处show diameter peers full all是从show support details获取的，它捕获了3gpp-aaa-s6b终端的Diameter对等体全部关闭的事实。请注意，这是从show support details(SSD)获取的show diameter peers full 命令的特殊调试版本，因此它还显示与aaamgr进程的所有对等连接（此处未显示输出）因此，连接的最终计数比正常运行时高得多，但底部显示的是摘要输出，就像它正常运行时连接数较少(144)一样。FULL输出附于本文，因此只显示一个对等体（但与所有6个双向代理）的连接，以便简单。

另示为Gy终端的一个开放工作连接的示例，您可以看到一个名为Local Address的额外字段，该字段捕获ASR端的连接处于启用状态，而在断开的3gpp-aaa-s6b对等体上，该字段不存在。（后面显示的是客户为包含本地地址的3gpp-aaa-s6b对等体修复问题后的输出。）

```

***** show diameter peers full *****
Sunday December 13 15:19:00 UTC 2020
-----
Context: billing Endpoint: 3gpp-aaa-s6b
-----

Peer Hostname: mp2.daladra01.dra.epc.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0001-diamproxy.s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org
Peer Realm: epc.mnc260.mcc310.3gppnetwork.org
Local Realm: epc.mnc260.mcc310.3gppnetwork.org

```

Peer Address: 10.160.113.136:3868
State: IDLE [TCP]
CPU: 10/0 Task: diamproxy-1
Messages Out/Queued: 0/0
Supported Vendor IDs: None
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A

Peer Hostname: mp2.daladra01.dra.epc.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0002-diamproxy.s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org
Peer Realm: epc.mnc260.mcc310.3gppnetwork.org
Local Realm: epc.mnc260.mcc310.3gppnetwork.org
Peer Address: 10.160.113.136:3868
State: IDLE [TCP]
CPU: 5/0 Task: diamproxy-2
Messages Out/Queued: 0/0
Supported Vendor IDs: None
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A

Peer Hostname: mp2.daladra01.dra.epc.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0003-diamproxy.s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org
Peer Realm: epc.mnc260.mcc310.3gppnetwork.org
Local Realm: epc.mnc260.mcc310.3gppnetwork.org
Peer Address: 10.160.113.136:3868
State: IDLE [TCP]
CPU: 7/0 Task: diamproxy-3
Messages Out/Queued: 0/0
Supported Vendor IDs: None
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A

Peer Hostname: mp2.daladra01.dra.epc.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0004-diamproxy.s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org
Peer Realm: epc.mnc260.mcc310.3gppnetwork.org
Local Realm: epc.mnc260.mcc310.3gppnetwork.org
Peer Address: 10.160.113.136:3868
State: IDLE [TCP]
CPU: 8/0 Task: diamproxy-4
Messages Out/Queued: 0/0
Supported Vendor IDs: None
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A

Peer Hostname: mp2.daladra01.dra.epc.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0005-diamproxy.s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org
Peer Realm: epc.mnc260.mcc310.3gppnetwork.org
Local Realm: epc.mnc260.mcc310.3gppnetwork.org
Peer Address: 10.160.113.136:3868
State: IDLE [TCP]
CPU: 3/0 Task: diamproxy-5
Messages Out/Queued: 0/0

```
Supported Vendor IDs: None
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A
```

```
Peer Hostname: mp2.daladra01.dra.epc.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0006-diamproxy.s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org
Peer Realm: epc.mnc260.mcc310.3gppnetwork.org
Local Realm: epc.mnc260.mcc310.3gppnetwork.org
Peer Address: 10.160.113.136:3868
State: IDLE [TCP]
CPU: 6/0 Task: diamproxy-6
Messages Out/Queued: 0/0
Supported Vendor IDs: None
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A
```

...

```
-----
Context: billing Endpoint: credit-control
-----
```

...

```
Peer Hostname: ln24.daladra01.dra.epc3.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0001-diamproxy.iepcf201.gy
Peer Realm: nsn-gy
Local Realm: starent.gy.com
Peer Address: 10.160.113.136:3869
Local Address: 10.168.86.151:55584
State: OPEN [TCP]
CPU: 10/0 Task: diamproxy-1
Messages Out/Queued: 0/0
Supported Vendor IDs: 10415
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A
```

```
Peers Summary:
Peers in OPEN state: 1404
Peers in CLOSED state: 468
Peers in intermediate state: 0
Total peers matching specified criteria: 1872
```

以下是此命令的正常输出，以供参考，显示没有aaamgr的连接计数：

```
Peers Summary:
Peers in OPEN state: 107
Peers in CLOSED state: 36
Peers in intermediate state: 1
Total peers matching specified criteria: 144
```

数据包捕获收集

如前所述，此场景显示s6b终端的所有直径对等体都处于关闭状态，问题不在于特定的diamproxy/card，这意味着任何卡的PCAP收集应适当地代表故障排除问题。如果仅在特定

Diamproxy上发现问题，则捕获该过程的PCAP将更为重要。这一点很重要，因为收集过程需要指定特定卡，而不能通过单次捕获在所有卡上运行，尽管在这种情况下，问题确实会在所有卡上出现，下面显示的是对两张卡的捕获，以帮助就如何分析结果数据提出一些看法。

首先，查看卡表，挑选几张要运行捕获的ACTIVE卡（3和5），并注明哪张是不应指定的解复用卡。

```
[local]IEPCF201# show card table
Friday December 11 17:15:28 UTC 2020
Slot          Card Type                                Oper State   SPOF  Attach
-----
1: CFC        Control Function Virtual Card             Active       No
2: CFC        Control Function Virtual Card             Standby      -
3: FC         4-Port Service Function Virtual Card     Active       No     <=====
4: FC         4-Port Service Function Virtual Card     Standby      -
5: FC         4-Port Service Function Virtual Card     Active       No     <=====
6: FC         4-Port Service Function Virtual Card     Active       No
7: FC         4-Port Service Function Virtual Card     Active       No
8: FC         4-Port Service Function Virtual Card     Active       No
9: FC         4-Port Service Function Virtual Card     Active       No
10: FC        4-Port Service Function Virtual Card     Active       No
[local]IEPCF201#
```

```
[local]IEPCF201# show session recovery status verbose
Saturday December 12 21:43:11 UTC 2020
Session Recovery Status:
  Overall Status      : Ready For Recovery
  Last Status Update  : 4 seconds ago

  ----sessmgr---  ----aaamgr----  demux
cpu state  active standby  active standby  active  status
-----
3/0 Active  12    1    12    1    0    Good
4/0 Standby 0    12   0    12   0    Good
5/0 Active  12    1    12    1    0    Good
6/0 Active  12    1    12    1    0    Good
7/0 Active  12    1    12    1    0    Good
8/0 Active  12    1    12    1    0    Good
9/0 Active  0     0    0     0    8    Good (Demux)
10/0 Active 12    1    12    1    0    Good
[local]IEPCF201#
```

此外，需要检索定义直径对等体的上下文#，在这种情况下，计费上下文为#2。

```
***** show context *****
Sunday December 13 15:14:24 UTC 2020
Context Name  ContextID  State  Description
-----
local         1          Active
billing       2          Active  <=====
calea        3          Active
gi           4          Active
sgw          5          Active
```

接下来，登录Linux调试外壳，以在其自己的CLI会话中收集PCAP的卡（本例中为卡3和5）：

注意：大多数操作员可能不会访问调试外壳，除非他们被告知密码，该密码特定于机箱/客户，具体取决于其设置方式。登录调试外壳时请小心，因为它登录到卡的底层操作系统（PSC或

ASR 5000或ASR 5500) 或虚拟机(vPC-DI的服务功能(SF))。

```
[local]IEPCF201# cli test password <password>
Saturday December 12 21:43:54 UTC 2020
Warning: Test commands enables internal testing and debugging commands
        USE OF THIS MODE MAY CAUSE SIGNIFICANT SERVICE INTERRUPTION
[local]IEPCF201#
[local]IEPCF201# debug shell card 3 cpu 0
Saturday December 12 21:44:02 UTC 2020
Last login: Fri Dec 11 19:26:34 +0000 2020 on pts/1 from card1-cpu0.
qvpc-di:card3-cpu0#
```

现在，请运行仅在此自定义StarOS版本的Linux中可用的特殊Linux命令setvr(set virtual router)，指定之前检索的上下文编号。请注意，提示符会更改：

```
qvpc-di:card3-cpu0# setvr 2 bash
bash-2.05b#
```

此时，可以使用以下参数运行TCP转储。请注意，如果端口号与前面gy示例中所示的端口号不同，则应使用该端口号。此外，如果有特定的对等地址要捕获数据包，则可以使用host <host ip address>指定主机IP地址。运行该命令几分钟，然后使用Control-C停止捕获。如果捕获了IF数据包，则显示数据包数。

```
bash-2.05b# tcpdump -i any -s 0 -w /tmp/diameter_SF3.pcap "port 3868"
tcpdump: listening on any
^C
1458 packets received by filter
0 packets dropped by kernel
bash-2.05b#
```

接下来，使用exit命令退出虚拟路由器空间，然后将文件复制到主用管理卡的闪存中，对于ASR 5500，闪存为MIO 5或6，对于vPC-DI，则此处为1或2。

```
bash-2.05b# exit
exit
qvpc-di:card3-cpu0# scp /tmp/diameter_SF3.pcap card1:/flash/sftp/diameter_SF3.pcap
diameter_SF3.pcap          100% 110KB 110.4KB/s   00:00
qvpc-di:card3-cpu0# exit
[local]IEPCF201#
```

此时，可使用sftp检索文件，使用网络中存在的任何方法访问/flash目录。

以下是SF 5的命令，这与刚才显示的SF 3的命令相同。理想情况下，同时运行两个会话，以便同时捕获以供分析（尽管这可能不是必需的）。

```
[local]IEPCF201# cli test password <password>
Saturday December 12 21:43:28 UTC 2020
Warning: Test commands enables internal testing and debugging commands
        USE OF THIS MODE MAY CAUSE SIGNIFICANT SERVICE INTERRUPTION
[local]IEPCF201# debug shell card 5 cpu 0
Saturday December 12 21:44:13 UTC 2020
qvpc-di:card5-cpu0#
qvpc-di:card5-cpu0# setvr 2 bash
bash-2.05b# tcpdump -i any -s 0 -w /tmp/diameter_SF5.pcap "port 3868"
tcpdump: listening on any
^C
1488 packets received by filter
```

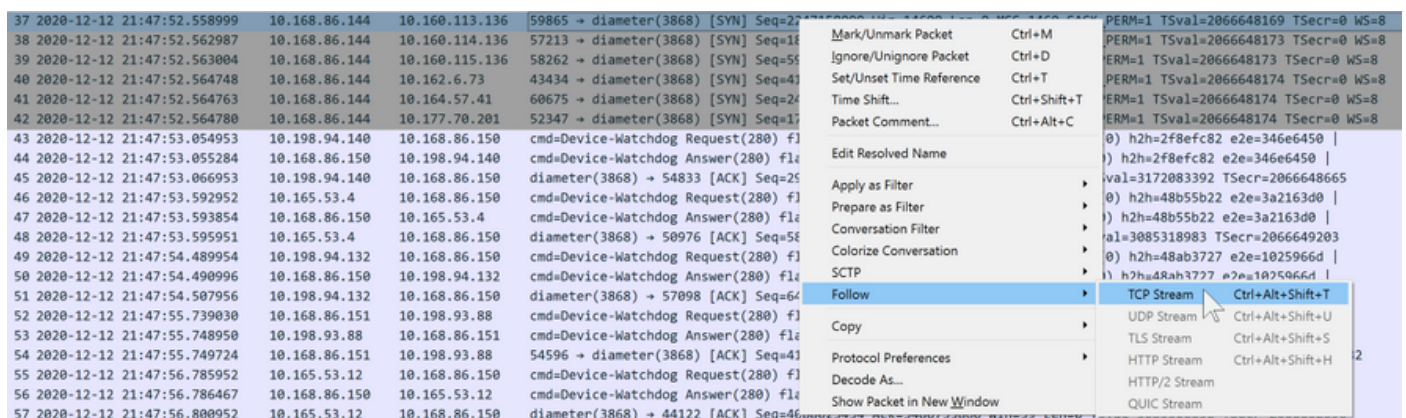
```

0 packets dropped by kernel
bash-2.05b# exit
exit
qvpc-di:card5-cpu0# scp /tmp/diameter_SF5.pcap card1:/flash/sftp/diameter_SF5.pcap
diameter_SF5.pcap          100% 113KB 112.7KB/s   00:00
qvpc-di:card5-cpu0# exit
[local]IEPCF201#

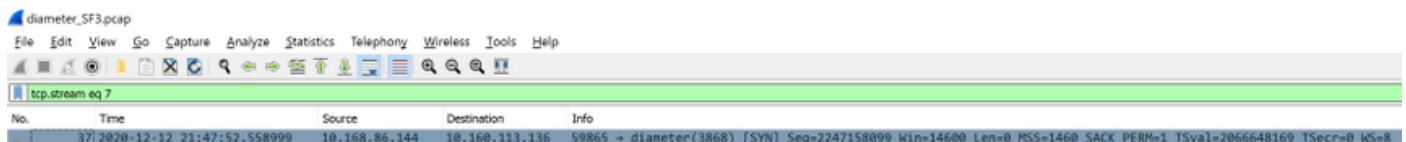
```

PCAP分析

此处的目标是确定在直径连接建立过程中故障的位置。如前所述，它可能在TCP/IP连接中，也可能在随后的CER/CEA步骤中。对于TCP/IP，查看是否正在发送TCP SYN以及是否正在接收TCP SYN ACK，然后是从ASR发送的ACK。可以使用任意数量的过滤器过滤数据包，以帮助分析。在这种情况下，过滤器tcp.flags.syn == 1显示SYN正针对此特定卡的所有6个对等体发送。查看未过滤的视图，右击SYN数据包并利用Wireshark中的TCP数据流功能，通过选择Follow ...（跟踪），该功能聚合了使用相同TCP端口号的所有TCP数据包。TCP数据流，查看是否存在建立连接的TCP数据包的对应交换。



在此场景中，请注意SYN之外没有其他数据包，这确认了ASR可能发送SYN但不会返回任何响应，这将消除ASR是连接设置失败的原因（虽然这不能保证是这样，可能不发送数据包，或者响应被丢弃，在这种情况下，外部PCAP会有帮助进一步缩小问题范围）。



另需注意的是，模式每30秒重复一次，与直径端点的默认配置30秒匹配以重试连接 — ASR不会放弃，而是会永远重试，直到成功。SF 5的PCAP显示完全相同的行为。

```

context billing
diameter endpoint 3gpp-aaa-s6b
connection timeout 30
connection retry-timeout 30

```


| No. | Time | Source | Destination | Info |
|-----|----------------------------|---------------|----------------|--|
| 37 | 2020-12-12 21:47:52.558999 | 10.168.86.144 | 10.160.113.136 | 59865 → diameter(3868) [SYN] Seq=2247158099 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066648169 TSecr=0 WS=8 |
| 38 | 2020-12-12 21:47:52.562987 | 10.168.86.144 | 10.160.114.136 | 57213 → diameter(3868) [SYN] Seq=1806187659 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066648173 TSecr=0 WS=8 |
| 39 | 2020-12-12 21:47:52.563004 | 10.168.86.144 | 10.160.115.136 | 58262 → diameter(3868) [SYN] Seq=593422692 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066648173 TSecr=0 WS=8 |
| 40 | 2020-12-12 21:47:52.564748 | 10.168.86.144 | 10.162.6.73 | 43434 → diameter(3868) [SYN] Seq=4111917603 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066648174 TSecr=0 WS=8 |
| 41 | 2020-12-12 21:47:52.564763 | 10.168.86.144 | 10.164.57.41 | 60675 → diameter(3868) [SYN] Seq=249946840 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066648174 TSecr=0 WS=8 |
| 42 | 2020-12-12 21:47:52.564780 | 10.168.86.144 | 10.177.70.201 | 52347 → diameter(3868) [SYN] Seq=171243962 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066648174 TSecr=0 WS=8 |
| 133 | 2020-12-12 21:48:22.592084 | 10.168.86.144 | 10.160.113.136 | 46954 → diameter(3868) [SYN] Seq=1599801985 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066678202 TSecr=0 WS=8 |
| 134 | 2020-12-12 21:48:22.592112 | 10.168.86.144 | 10.160.114.136 | 35751 → diameter(3868) [SYN] Seq=3337865783 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066678202 TSecr=0 WS=8 |
| 135 | 2020-12-12 21:48:22.592129 | 10.168.86.144 | 10.160.115.136 | 43169 → diameter(3868) [SYN] Seq=3026367013 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066678202 TSecr=0 WS=8 |
| 136 | 2020-12-12 21:48:22.592143 | 10.168.86.144 | 10.162.6.73 | 59796 → diameter(3868) [SYN] Seq=1603160447 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066678202 TSecr=0 WS=8 |
| 137 | 2020-12-12 21:48:22.599364 | 10.168.86.144 | 10.164.57.41 | 60677 → diameter(3868) [SYN] Seq=3877471182 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066678209 TSecr=0 WS=8 |
| 138 | 2020-12-12 21:48:22.599396 | 10.168.86.144 | 10.177.70.201 | 50877 → diameter(3868) [SYN] Seq=375168575 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066678209 TSecr=0 WS=8 |
| 217 | 2020-12-12 21:48:52.595089 | 10.168.86.144 | 10.160.113.136 | 47032 → diameter(3868) [SYN] Seq=3396628935 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066708205 TSecr=0 WS=8 |
| 218 | 2020-12-12 21:48:52.595110 | 10.168.86.144 | 10.160.114.136 | 33418 → diameter(3868) [SYN] Seq=1405313703 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066708205 TSecr=0 WS=8 |
| 219 | 2020-12-12 21:48:52.596989 | 10.168.86.144 | 10.160.115.136 | 37717 → diameter(3868) [SYN] Seq=4103832795 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066708207 TSecr=0 WS=8 |
| 220 | 2020-12-12 21:48:52.597006 | 10.168.86.144 | 10.162.6.73 | 43508 → diameter(3868) [SYN] Seq=1142592045 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066708207 TSecr=0 WS=8 |
| 221 | 2020-12-12 21:48:52.597024 | 10.168.86.144 | 10.164.57.41 | 32922 → diameter(3868) [SYN] Seq=1673081762 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066708207 TSecr=0 WS=8 |
| 222 | 2020-12-12 21:48:52.597038 | 10.168.86.144 | 10.177.70.201 | 38623 → diameter(3868) [SYN] Seq=2074222018 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066708207 TSecr=0 WS=8 |
| 313 | 2020-12-12 21:49:22.614018 | 10.168.86.144 | 10.160.113.136 | 37338 → diameter(3868) [SYN] Seq=1371056611 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066738224 TSecr=0 WS=8 |
| 314 | 2020-12-12 21:49:22.614045 | 10.168.86.144 | 10.160.114.136 | 43483 → diameter(3868) [SYN] Seq=4212342380 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066738224 TSecr=0 WS=8 |
| 315 | 2020-12-12 21:49:22.616176 | 10.168.86.144 | 10.160.115.136 | 60092 → diameter(3868) [SYN] Seq=2954594158 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066738226 TSecr=0 WS=8 |
| 316 | 2020-12-12 21:49:22.616196 | 10.168.86.144 | 10.162.6.73 | 34616 → diameter(3868) [SYN] Seq=332280458 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066738226 TSecr=0 WS=8 |
| 317 | 2020-12-12 21:49:22.616211 | 10.168.86.144 | 10.164.57.41 | 52412 → diameter(3868) [SYN] Seq=1830555143 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066738226 TSecr=0 WS=8 |
| 318 | 2020-12-12 21:49:22.616228 | 10.168.86.144 | 10.177.70.201 | 44325 → diameter(3868) [SYN] Seq=2745428018 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066738226 TSecr=0 WS=8 |
| 406 | 2020-12-12 21:49:52.620143 | 10.168.86.144 | 10.160.113.136 | 57729 → diameter(3868) [SYN] Seq=52777398 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066768230 TSecr=0 WS=8 |
| 407 | 2020-12-12 21:49:52.621217 | 10.168.86.144 | 10.160.114.136 | 53024 → diameter(3868) [SYN] Seq=3814405758 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066768231 TSecr=0 WS=8 |
| 408 | 2020-12-12 21:49:52.621235 | 10.168.86.144 | 10.160.115.136 | 53651 → diameter(3868) [SYN] Seq=593445658 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066768231 TSecr=0 WS=8 |
| 409 | 2020-12-12 21:49:52.621248 | 10.168.86.144 | 10.162.6.73 | 57360 → diameter(3868) [SYN] Seq=3087447500 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066768231 TSecr=0 WS=8 |
| 410 | 2020-12-12 21:49:52.625336 | 10.168.86.144 | 10.164.57.41 | 34845 → diameter(3868) [SYN] Seq=560819250 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066768235 TSecr=0 WS=8 |
| 411 | 2020-12-12 21:49:52.625353 | 10.168.86.144 | 10.177.70.201 | 44899 → diameter(3868) [SYN] Seq=2172486101 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066768235 TSecr=0 WS=8 |
| 505 | 2020-12-12 21:50:22.637579 | 10.168.86.144 | 10.160.113.136 | 55966 → diameter(3868) [SYN] Seq=3186446422 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066798247 TSecr=0 WS=8 |
| 506 | 2020-12-12 21:50:22.639702 | 10.168.86.144 | 10.160.114.136 | 39076 → diameter(3868) [SYN] Seq=3286959003 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066798249 TSecr=0 WS=8 |

将事物连在一起，直径基本统计数据显示，失败连接的数量以与SF/diamproxies数量和重试超时相称的速率递增。计算方法如下：6个对等体* 6个diamproxies =每30秒尝试36次。因此，在一分钟内将有72次尝试，这可以通过运行**show diameter statistics proxy**和查看**Connection Timeouts** (连接超时从60984增加到61056 = 72)来看到，如CLI时间戳所示。

```
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:10 UTC 2020
Connection Timeouts: 60984
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:12 UTC 2020
Connection Timeouts: 60984
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:14 UTC 2020
Connection Timeouts: 60984
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:17 UTC 2020
Connection Timeouts: 60990
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:19 UTC 2020
Connection Timeouts: 60990
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:21 UTC 2020
Connection Timeouts: 60996
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:25 UTC 2020
Connection Timeouts: 61002
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:27 UTC 2020
Connection Timeouts: 61002
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:29 UTC 2020
Connection Timeouts: 61008
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:32 UTC 2020
Connection Timeouts: 61014
[local]IEPCF201# show diameter statistics proxy | grep "Connection Timeouts"
Friday December 11 20:39:35 UTC 2020
Connection Timeouts: 61014
```


另请注意，CER/CEA（跨所有直径对等体）的数量很少，这证明它永远无法尝试交换这些数据包，这意味着这是TCP/IP设置问题。

```
[local]IEPCF201# show diameter statistics proxy
Friday December 11 20:57:09 UTC 2020
...
Capabilities Exchange Requests and Answers statistics:
  Connection CER sent:                               109
  Connection CER send errors:                         0
  CERs received:                                     0
  Connection CER create failures:                     0
  CEAs received:                                     108
  CEA AVPs unknown:                                  0
  CEA Application ID mismatch:                       0
  Read CEA Messages:                                 108
  Read CEA Messages Unexpected:                       0
  Read CEA Missing:                                  0
  Read CEA Negotiation Failure:                     0
  Read CER Messages:                                 0
  Read CER Messages Unexpected:                       0
  Read CER Missing:                                  0
  Tw Expire Waiting for CEA:                         0
```

最后请注意，在客户解决问题后，处于CLOSED状态的对等体将返回0,Local Address字段显示在show diameter peers中，全部输出。

```
Peer Hostname: mp1.daldra01.dra.epc.mnc260.mcc310.3gppnetwork.org
Local Hostname: 0001-diamproxy.s6b.IEPCF201.epc.mnc260.mcc310.3gppnetwork.org
Peer Realm: epc.mnc260.mcc310.3gppnetwork.org
Local Realm: epc.mnc260.mcc310.3gppnetwork.org
Peer Address: 10.160.113.133:3868
Local Address: 10.168.86.144:32852
State: OPEN [TCP]
CPU: 10/0                                     Task: diamproxy-1
Messages Out/Queued: 0/0
Supported Vendor IDs: None
Admin Status: Enable
DPR Disconnect: N/A
Peer Backoff Timer running:N/A
```

```
Peers Summary:
  Peers in OPEN state: 144
  Peers in CLOSED state: 0
  Peers in intermediate state: 0
  Total peers matching specified criteria: 144
```

```
[local]IEPCF101#
```