

Despejo de TCP Linux na shell de depuração para solucionar problemas de conexão de parâmetro

Contents

[Introduction](#)

[Informações de Apoio](#)

[Problemas de peering](#)

[Coleta de Captura de Pacotes](#)

[Análise de PCAP](#)

Introduction

Este documento descreve como usar o TCP Dump no shell de depuração do StarOS para solucionar problemas de conexão de Diameter. Frequentemente, são levantados casos solicitando assistência na solução de problemas por que uma conexão de Diameter não será ativada ou terá sido desativada, mesmo que (supostamente) nenhuma alteração na configuração ou na rede tenha ocorrido. A conexão de diâmetro pode falhar ao estabelecer no nível de negociação inicial do TCP/IP ou no nível de Solicitação de Troca de Recursos (CER - Capabilities Exchange Request)/Resposta de Troca de Recursos (CEA - Capabilities Exchange Answer).

Informações de Apoio

Embora não haja um problema típico de peering de diâmetro, eles se encaixam em algumas categorias:

- Todos os correspondentes de um endpoint e/ou protocolo específico estão inoperantes.
<===== este exemplo
- Os pares de um número de porta específico estão inoperantes.
- Os colegas vinculados a uma placa específica de PSC, DPC ou SF estão inoperantes.

Normalmente, a porta TCP 3868 (padrão) é usada para o lado do servidor de Diâmetro, embora outras portas também possam ser especificadas e confirmadas como diferentes de 3868 na configuração se as linhas de configuração do peer tiverem um número de porta especificado no final da linha.

Problemas de peering

No exemplo aqui, os peers para o ponto final 3gpp-aaa-s6b foram relatados por **show diâmetro peer full all** e não têm nenhum número de porta especificado nas linhas de peer e, portanto, por padrão, usam a porta 3868, enquanto os peers para Gy usam uma combinação de 3868, 3869 e 3870 para várias peers.

mostrar peers de diâmetro todos relatam todos os peers configurados para todos os endpoints de diâmetro. Aqui vemos 6 pares configurados e as linhas de configuração associadas para 3gpp-

aaa-s6b (quebrado) assim como para Gy (em funcionamento), observando que Gy tem alguns números de porta personalizados:

```
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.daladra01.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
```

Também é importante observar que, para a maioria das configurações, o usp-proxy configurável é especificado para configurar peering no lado ASR para usar o processo diamproxy em execução em todas as placas ativas, por exemplo, este é um vPC-DI em que as placas são chamadas Service Function Cards.

```

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

```

cpu	facility	task		cputime		memory		files		sessions		
		inst	used	allc	used	alloc	used	allc	used	allc	S	status
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#

```

Aqui, **mostrar os pares de diâmetro completos**, tudo é retirado do comando show support details captura o fato de que os correspondentes de Diameter para o endpoint 3gpp-aaa-s6b estão todos inativos. Observe que esta é uma versão de depuração especial do **comando show diâmetro peers full** extraído do comando show support details (SSD) e, portanto, também mostra todas as conexões de peer para os processos **aaamgr** (não mostrando a saída aqui) e, portanto, a

contagem final de conexões é muito maior do que se isso fosse executado normalmente, mas mostrado na parte inferior é a saída resumida como se ela fosse executada normalmente com o menor número de conexões (14) 4). A saída FULL está anexada a este artigo e, portanto, apenas as conexões de um peer (mas com todos os 6 diamproxies) são mostradas para ser breve.

Também é mostrado um exemplo de uma conexão de trabalho aberta para os endpoints Gy, onde você pode ver um campo extra chamado **Endereço local** que captura a conexão que está ativa no lado ASR, enquanto nos peers 3gpp-aaa-s6b quebrados esse campo não existe. (Mais tarde mostrado é a saída depois que o problema foi corrigido pelo cliente para o peer 3gpp-aaa-s6b onde esse **endereço local** está incluído.)

```
***** 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.daldra01.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.daldra01.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.daldra01.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.daldra01.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

Para referência, aqui está a saída normal desse comando mostrando a contagem de conexões sem os aaamgrs:

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

Coleta de Captura de Pacotes

Conforme discutido, este cenário mostra que TODOS os peers de diâmetro estão inativos para o endpoint s6b, o problema NÃO é para um diamproxy/placa específica, o que significa que a coleta PCAP para qualquer uma das placas deve representar adequadamente o problema para fins de solução de problemas. Se o problema fosse visto apenas em um proxy de diamante específico, então seria mais importante capturar um PCAP para esse processo. Isso é importante porque o processo de coleta requer a especificação de uma placa específica - ela não pode ser executada em todas as placas com uma única captura - e, embora nesse cenário o problema seja realmente visto em todas as placas, as capturas mostradas abaixo são realizadas em duas placas para ajudar a destacar como analisar os dados resultantes.

A primeira coisa a fazer é olhar para a tabela de placas e escolher algumas placas ATIVAS (3 e 5) nas quais executar a captura, além de observar qual é a placa Demux que não deve ser especificada.

```
[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#

```

Além disso, o número de contexto onde os peers de diâmetro são definidos precisa ser recuperado. Nesse caso, o contexto de cobrança é #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

```

Em seguida, faça login no shell de depuração do Linux para as placas onde o PCAP deve ser coletado, neste caso, as placas 3 e 5, em sua própria sessão CLI:

Note: O acesso ao shell de depuração não é algo ao qual a maioria dos operadores provavelmente terá acesso, a menos que lhes tenha sido comunicada a senha específica do chassi/cliente, dependendo de como foi configurado. Tenha cuidado ao fazer login no shell de depuração, pois ele está fazendo login no sistema operacional subjacente da placa (PSC ou DPC do ASR 5000 ou ASR 5500) ou máquina virtual (Service Function (SF) do vPC-DI).

```

[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#

```

Agora, execute um **setvr** de comando Linux especial (set virtual router) disponível apenas nesta versão personalizada do StarOS do Linux, especificando o contexto # recuperado anteriormente. Observe que o prompt muda:

```

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

```

Neste ponto, o dump TCP pode ser executado usando os parâmetros da seguinte maneira. Observe que se o número da porta for diferente do mostrado no exemplo anterior para gy, esse número de porta deve ser usado. Além disso, um endereço IP de host pode ser especificado com **host <host ip address>** se houver um endereço peer específico para o qual capturar pacotes. Execute o comando por alguns minutos e pare a captura com o Control-C. Se os pacotes forem

capturados, o número de pacotes será exibido.

```
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#
```

Em seguida, saia do espaço do roteador virtual com o comando exit e copie o arquivo para a flash da placa de gerenciamento ativa, que para o ASR 5500 seria MIO 5 ou 6, ou no caso aqui para vPC-DI, 1 ou 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#
```

Nesse ponto, o arquivo pode ser recuperado com sftp usando qualquer meio existente na rede para acessar o diretório /flash.

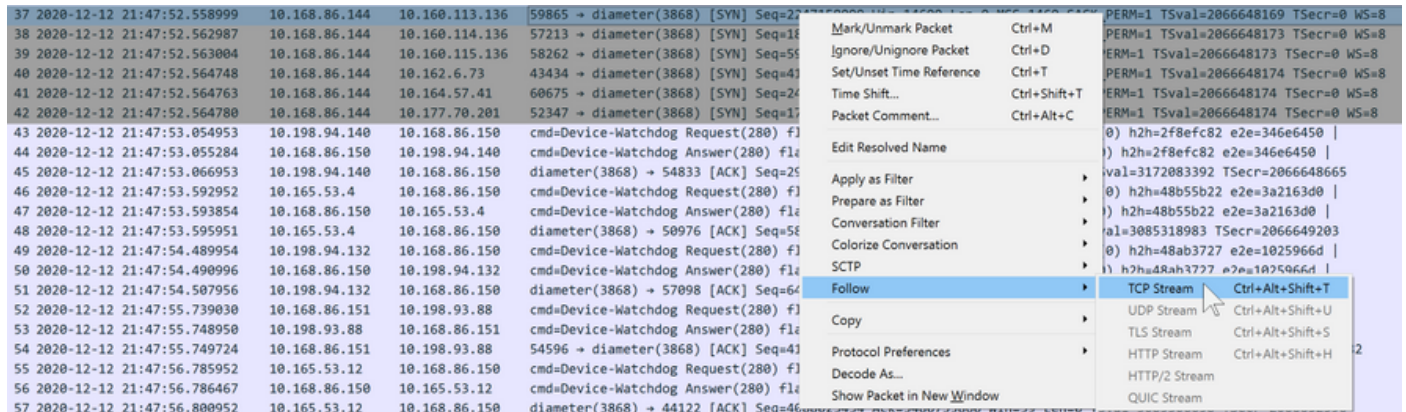
Aqui estão os comandos para SF 5 também, o que é uma repetição do que foi mostrado para SF 3. Idealmente, execute ambas as sessões ao mesmo tempo para ter capturas simultâneas para análise (embora isso possa não ser necessário).

```
[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#
```

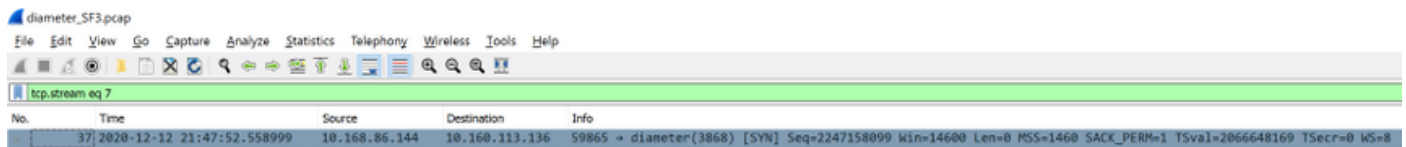
Análise de PCAP

O objetivo aqui é determinar onde está o detalhamento no processo de estabelecimento da conexão de diâmetro. Como mencionado anteriormente, ele pode estar na conexão TCP/IP ou na etapa CER/CEA subsequente. Para TCP/IP, verifique se um TCP SYN está sendo enviado e se um TCP SYN ACK está sendo recebido, seguido por um ACK enviado do ASR. Os pacotes podem ser filtrados com qualquer número de filtros para ajudar na análise e, nesse caso, o filtro `tcp.flags.syn == 1` mostra que o SYN está sendo enviado para todos os 6 peers desta placa específica. Olhando para uma exibição não filtrada, clique com o botão direito do mouse em um pacote SYN e aproveite o recurso de fluxo TCP no Wireshark que agrega todos os pacotes TCP

que estão usando o mesmo número de porta TCP, escolhendo Seguir ... Fluxo TCP para ver se há uma troca correspondente de pacotes TCP que estabelece a conexão.



Neste cenário, observe que NÃO há mais pacotes além do SYN, e isso confirma que o ASR provavelmente está enviando um SYN, mas não obtendo resposta, o que eliminaria o ASR de ser a causa da falha na configuração da conexão (embora não seja garantido que seja o caso, possivelmente o pacote não está sendo enviado ou que a resposta está sendo descartada, caso em que um PCAP externo seria útil para reduzir ainda mais o problema).



Também vale a pena observar que o padrão está sendo repetido a cada 30 segundos, o que corresponde à configuração padrão para o ponto final de diâmetro de 30 segundos para tentar novamente a conexão - o ASR não está desistindo, mas tentará para sempre até o sucesso. O PCAP para SF 5 mostra exatamente o mesmo comportamento.

```
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	10.160.113.136	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	10.160.113.136	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	10.160.113.136	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	10.160.113.136	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 21:48:22.59364	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 21:48:22.593996	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	10.160.113.136	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	10.160.113.136	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	10.160.113.136	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	10.160.113.136	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	10.160.113.136	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	10.160.113.136	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	10.160.113.136	39076 → diameter(3868) [SYN] Seq=3286959003 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2066798249 TSecr=0 WS=8

Amarrar as coisas, as estatísticas da base de diâmetro mostram que o número de conexões com falha está aumentando a uma taxa proporcional ao número de SF/diamproxies e o tempo limite de repetição. A matemática é a seguinte: $6 \text{ peers} * 6 \text{ diamproxies} = 36 \text{ tentativas a cada } 30 \text{ segundos}$. Então, em um minuto, seriam 72 tentativas, e isso pode ser visto executando **proxy de estatísticas de show diâmetro** e observando **Timeouts de Conexão** aumentando de 60984 para 61056 = 72 em um período de minuto, como mostrado pelos timestamps de CLI.

```
[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
```


Friday December 11 20:40:27 UTC 2020

Connection Timeouts:

61074

[local]IEPCF201#

Observe também que o número de CER/CEAs (em todos os pares de diâmetro) é trivial, o que prova que ele nunca chega ao ponto de tentar trocar esses pacotes, o que significa que é um problema de configuração de TCP/IP.

[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

Por fim, observe que depois que o problema foi resolvido pelo cliente, os Peers em CLOSED State retorna para 0 e o campo **Local Address** aparece em **show diâmetro peers full all output**.

```
Peer Hostname: mpl.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: 10.160.113.133:3868
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#