

Create Dynamic Service Flows (UGS) on CMTS with Test Command

Contents

[Introduction](#)

[Prerequisites](#)

[Requirement](#)

[Components Used](#)

[Background Information](#)

[Create The Flows](#)

[Upstream Direction](#)

[Decode of the TLVs](#)

[Downstream Direction](#)

[Decode of the TLVs](#)

[Delete the Flows](#)

[Related Information](#)

Introduction

This document describes the procedure and commands to create dynamic service flows on a Cable Modem Termination System (CMTS). For example, Unsolicited Grant Service (UGS) used in voice calls.

Prerequisites

Requirement

Cisco recommends that you have knowledge of these topics:

- CMTS
- DOCSIS

Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

In order to create a dynamic service flow (whatever the type), either the CMTS or the Cable

Modem (CM) need to send a Dynamic Service Add (DSA) message.

The DSA message contains two things:

- The Service Flow (SF) itself.
- The associated classifier(s).

The DSA message is a TLV encoded message which uses the same TLV definitions like the ones used for the CM.

Message sequence from PacketCable Multimedia (PCMM) specification:

10.2 Detailed Message Sequence

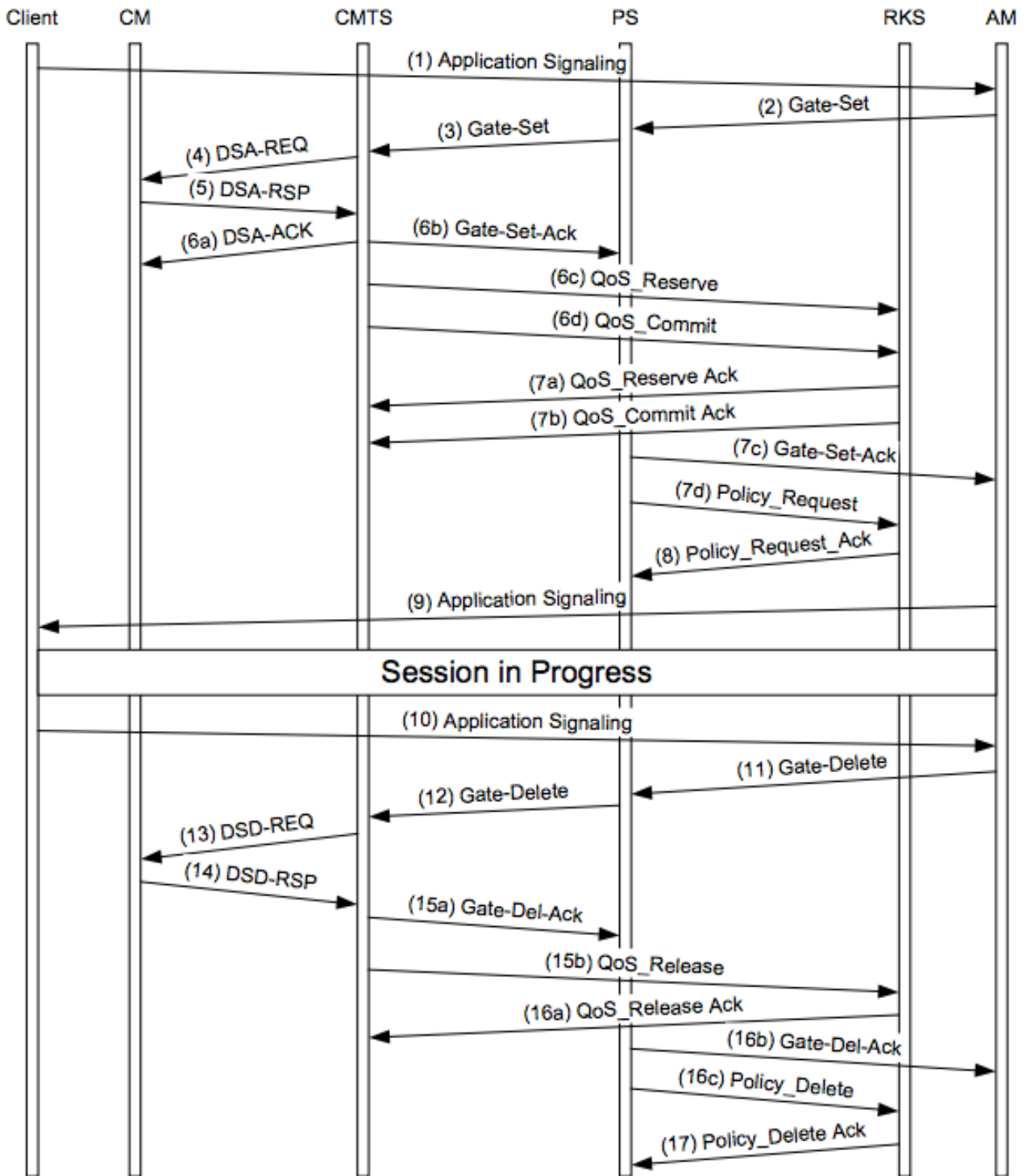


Figure 9 - Detailed Message Sequence

Create The Flows

In order to create the flow, you need to send a DSA per service flow that you want to create (you could potentially merge the 2 messages into a single message as well, but split them makes them easier to understand).

Note: the commands used here are for cBR8 platform. The uBR10k uses the same command syntax, but without the docsis keyword in the test commands

Upstream Direction

Example:

```
test cable docsis dsa c005.c22c.dc5b message
1823010200040601060F010610040000037F130200E8160101150400000320140400004E20160f010102030200040906
05040a305879
```

Decode of the TLVs

1. Service Flow TLV.

The Upstream SF TLV is TLV 24 (0x18):

```
1823010200040601060F010610040000037F130200E8160101150400000320140400004E20
```

Decode:

```
1823 TLV 0x18 (24) - Len 35 - Upstream Service Flow (0x23 = 35)
01020004 TLV 0x18.1 (24.1) - Len 02 - Upstream Service Flow ID - value 0004
060106 TLV 0x18.6 (24.6) - Len 01 - QoS Parameter set type - value 06 -> Admitted and Active bit
set
0F0106 TLV 0x18.F (24.15) - Len 01 - Service flow scheduling type - value 06 -> UGS
10040000037F TLV 0x18.10(24.16) - Len 04 - Request Transmit policy - value 0x0000037F
130200E8 TLV 0x18.13(24.19) - Len 02 - Number of bytes - value 0xe8 -> 232 bytes
160101 TLV 0x18.16(24.22) - Len 01 - Number of grants per interval- value 1
150400000320 TLV 0x18.15(24.21) - Len 04 - Tolerated Grant Jitter - value 0x320 -> 800
140400004E20 TLV 0x18.14(24.20) - Len 04 - Nominal Grant interval - value 0x4e20 -> 20000
```

2. Classifier TLV.

The classifier used in this example is a simple IP packet classifier only. Other types of classifier exist (Protocol, UDP/TCP, etc.).

The Upstream Classifier TLV is TLV 22 (0x16):

```
160f01010203020004090605040a305879
```

Decode:

```
160f TLV 0x16 (22) - Len 15 - Upstream Classifier
010102 TLV 0x16.1 (22.1) - Len 01 - Classifier reference - Unique classifier ID - value 0x02
03020004 TLV 0x16.2 (22.2) - Len 02 - Service flow reference - value 0004 -> MUST match the SFID
above
0906 TLV 0x16.9 (22.9) - Len 06 - IPv4 packet classifier encoding
05040a305879 TLV 0x16.9.5(22.9.5) - Len 04 - Destination IPv4 classifier - value 0a305879
10.48.88.121
```

It corresponds to this service flow/classifier on the CM configuration file:

```
160f01010203020004090605040a305879
```

Decode:

```
160f TLV 0x16 (22) - Len 15 - Upstream Classifier
010102 TLV 0x16.1 (22.1) - Len 01 - Classifier reference - Unique classifier ID - value 0x02
```

03020004 TLV 0x16.2 (22.2) - Len 02 - Service flow reference - value 0004 -> MUST match the SFID above

0906 TLV 0x16.9 (22.9) - Len 06 - IPv4 packet classifier encoding

05040a305879 TLV 0x16.9.5(22.9.5) - Len 04 - Destination IPv4 classifier - value 0a305879 10.48.88.121

Downstream Direction

Example:

```
test cable docsis dsa c005.c22c.dc5b message
```

```
191A010200990601060701050804000154A00A04000154A00B0200DA170f01010203020099090603040a305879
```

Decode of the TLVs

1. Service Flow TLV.

The Downstream Service Flow TLV is TLV 25 (0x19):

```
191A010200990601060701050804000154A00A04000154A00B0200DA
```

Decode:

191A TLV 0x19 (25) - Len 1A (26) - Downstream Service Flow definition

01020099 TLV 0x19.1 (25.1) - Len 02 - Downstream Service Flow ID - value 0x99

060106 TLV 0x19.6 (25.6) - Len 01 - QoS Parameter set type - value 06 -> Admitted and Active bit set

070105 TLV 0x19.7 (25.7) - Len 01 - Traffic Priority - value 05 -> Prio 5

0804000154A0 TLV 0x19.8 (25.8) - Len 04 - Max Sustain Rate - value 0x154a0 = 87200 bps

0A04000154A0 TLV 0x19.A (25.10) - Len 04 - Min Reserved Rate - value 0x154a0 = 87200 bps

0B0200DA TLV 0x19.B (25.11) - Len 02 - Assumed Min Rvd Rate packet size - value 0xda = 218

2. Classifier TLV.

This is also a plain IPv4 classifier. A more complex classifier can also be created.

The Downstream Classifier TLV is TLV 23 (0x17):

```
170f01010203020099090603040a305879
```

Decode:

170f TLV 0x17 (23) - Len 15 - Downstream Classifier

010102 TLV 0x17.1 (23.1) - Len 01 - Downstream Classifier Reference - value 0x02

03020099 TLV 0x17.3 (23.3) - Len 02 - Downstream Service Flow ID reference - value 0x99 -> MUST match SFID above

0906 TLV 0x17.9 (23.9) - Len 06 - IPv4 classifier

03040a305879 TLV 0x17.9.3(23.9.3) - Len 04 - Source IPv4 Address - value 0x0a305879 -> 10.48.88.121

It corresponds to this service flow/classifier on the CM configuration file:

```
170f01010203020099090603040a305879
```

Decode:

170f TLV 0x17 (23) - Len 15 - Downstream Classifier

010102 TLV 0x17.1 (23.1) - Len 01 - Downstream Classifier Reference - value 0x02

03020099 TLV 0x17.3 (23.3) - Len 02 - Downstream Service Flow ID reference - value 0x99 -> MUST match SFID above

0906 TLV 0x17.9 (23.9) - Len 06 - IPv4 classifier

03040a305879 TLV 0x17.9.3(23.9.3) - Len 04 - Source IPv4 Address - value 0x0a305879 -> 10.48.88.121

Delete the Flows

Dynamic service flows can be deleted with a Dynamic-Service-Delete (DSD) message. The procedure is identical to delete both US SF and DS SF.

test cable docsis dsd <mac> <service-flow-id>

Example:

acdc-cbr8-2#show cable modem 2cab.a40c.5598 service-flow

SUMMARY:

MAC Address	IP Address	Host Interface	MAC State	Prim Sid	Num CPE	Primary Downstream	DS RfId
2cab.a40c.5598	172.54.0.4	C1/0/2/UB	w-online	27	0	In1/0/2:7	8711

Sfid	Dir	Curr State	Sid	Sched Type	Prio	MaxSusRate	MaxBrst	MinRsvRate	Throughput
17	US	act	27	BE	5	1024	3044	0	929
18	DS	act	N/A	N/A	3	300000000	24600	0	887

--> Before : 2SFs only

acdc-cbr8-2#test cable docsis dsa 2cab.a40c.5598 message

1823010200040601060F010610040000037F130200E8160101150400000320140400004E20160f01010203020004090605040a305879

--> UGS SF

acdc-cbr8-2#test cable docsis dsa 2cab.a40c.5598 message

191A010200990601060701050804000154A00A04000154A00B0200DA170f01010203020099090603040a305879

--> DS SF

acdc-cbr8-2#show cable modem 2cab.a40c.5598 service-flow

Load for five secs: 10%/1%; one minute: 9%; five minutes: 10%

Time source is NTP, 10:54:57.426 CET Thu Nov 22 2018

SUMMARY:

MAC Address	IP Address	Host Interface	MAC State	Prim Sid	Num CPE	Primary Downstream	DS RfId
2cab.a40c.5598	172.54.0.4	C1/0/2/UB	w-online	27	0	In1/0/2:7	8711

Sfid	Dir	Curr State	Sid	Sched Type	Prio	MaxSusRate	MaxBrst	MinRsvRate	Throughput
17	US	act	27	BE	5	1024	3044	0	896
57	US	act	43	UGS	0	0	0	0	0
18	DS	act	N/A	N/A	3	300000000	24600	0	0
58	DS	act	N/A	N/A	5	87200	3044	87200	0

--> Now Both UGS and DS voice flows are created (and throughput would be seen as soon as packets

match the classifier.)

Related Information

- [PacketCable Multimedia Specification](#)
- [Technical Support & Documentation - Cisco Systems](#)