

使用cBR-8、TSDuck和VLC配置DVB-C實驗室環境

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[設定](#)

[cBR-8視訊作業階段](#)

[串流器](#)

[ECMG](#)

[驗證](#)

[在cBR-8上](#)

[關於ECMG](#)

[疑難排解](#)

[相關資訊](#)

簡介

本文說明如何使用TSDuck工具包、VLC和cBR-8配置數位影片廣播 — 電纜(DVB-C)實驗場景。

必要條件

需求

思科建議您瞭解以下主題：

- DVB-C
- Symulcrypt
- VoD
- cBR-8

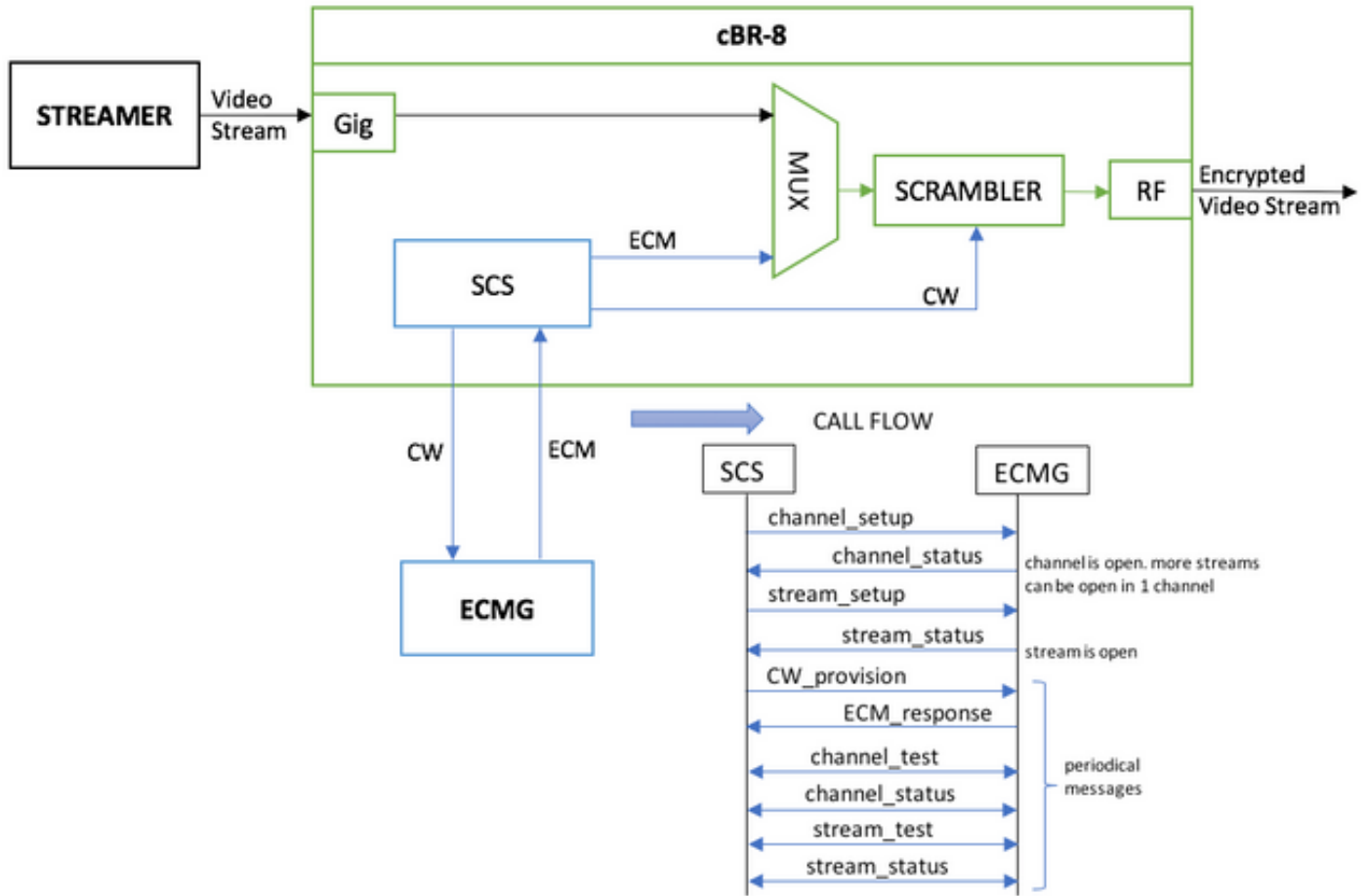
採用元件

本文件所述內容不限於特定軟體和硬體版本。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

背景資訊

本文檔中顯示的場景（如下圖所示）涉及cBR-8 as iCMTS、使用VLC作為影片流處理器的Linux虛擬機器(VM)和使用TSDuck的Linux虛擬機器。重新建立DVB-Symulcrypt加密系統，其中cBR8充當模擬加密同步器(SCS)，而TSDuck VM扮演權利控制消息生成器(ECMG)的角色，就像它是Nagra伺服器一樣。



充當流處理器的VM只傳送本地儲存的影片片段，該影片片段會循環以模擬連續流。cBR-8為此模擬配置了一個基於表（靜態）會話，並且沒有請求VoD流的機頂盒(STB)或數據機，它是在流器上手動啟動的。

收到資料流時，cBR-8嘗試與配置的ECMG伺服器通訊，以便加密影片流，並交換上圖呼叫流中所述的消息。這些報文將與TSDuck進行清晰交換，TSDuck有助於分析報文和調試的內容。而且TSDuck會回覆所有已傳送的請求，而不檢查引數（ca-system-id、access-criteria等）的正確性。

如果cBR-8無法與ECMG通訊，則由於指令無法清除而清除地傳送流。

在真實情況下，需要向STB傳送授權管理消息(EMM)，授權接收者解密特定控制字(CW)。EMM可以通過cBR-8或單獨的通道傳送到接收器，TSDuck還具有模擬EMM生成器(EMMG)的功能

設定

cBR-8視訊作業階段

以下示例說明如何在cBR-8上配置DVB影片會話。訪問標準通常由條件訪問系統(CAS)提供，在此模擬情況下，您可以生成隨機十六進位制數以及ca-system-id。

virtual-edge-input-ip是流的IP目標，在這種情況下，這不是實際目標，但必須是用於從流傳送影片流的IP。

```

cable video
  encryption
    linecard 1/0 ca-system dvb scrambler dvb-csa
    dvb
      ecmg NAGRA_ELK id 1
        mode tier-based
        type nagra
        ca-system-id 2775 3
        auto-channel-id
        ecm-pid-source auto 48 8190
        connection id 1 priority 1 10.48.88.12 3337
        overrule
          min-cp-duration 300000
        tier-based
          ecmg name NAGRA_ELK access-criteria c972bfd7701e6d28069ae85f5d701d63ac1aec4a
          fail-to-clear
          enable
      service-distribution-group SDG-ACDC-LAB-TEST1 id 1
        onid 100
        rf-port integrated-cable 1/0/3
    virtual-carrier-group VCG-ACDC-LAB-TEST1 id 1
      encrypt
      service-type narrowcast
      rf-channel 32-35 tsid 42496-42499 output-port-number 1-4
    bind-vcg
      vcg VCG-ACDC-LAB-TEST1 sdg SDG-ACDC-LAB-TEST1
    logical-edge-device LED-ACDC-LAB-TEST1 id 1
      protocol table-based
        virtual-edge-input-ip 10.10.10.10 input-port-number 1
        vcg VCG-ACDC-LAB-TEST1
        active
      table-based
        vcg VCG-ACDC-LAB-TEST1
        rf-channel 32
          session vod1 input-port 1 start-udp-port 65 num-sessions-per-qam 1 processing-type remap
    start-program 1
  !
controller Integrated-Cable 1/0/3
  max-carrier 44
  base-channel-power 40
  rf-chan 32 35
  type VIDEO
  frequency 850000000
  rf-output NORMAL
  power-adjust 0.0
  qam-profile 3

```

串流器

在此裝置上，您只需從命令列安裝VLC，然後啟動本地儲存的影片檔案流。
請參考官方[文檔](#)。

安裝VLC後，以下命令列顯示如何啟動名為cisco-tac-lab.mov的檔案的流，指定目標IP和埠、cBR-8上的tsid和埠，並循環影片以模擬連續流(—repeat):

```

cvlc cisco-tac-lab.mov —sout
'#duplicate{dst=udp{mux=ts , dst=10.10.10.10:65,tsid=42496,port=65}' — 重複和

```

ECMG

從官方網站下載TSDuck:[TSDuck](https://tsduck.net/)，並參考使用手冊文檔，以便安裝和查詢功能資訊。

安裝TSDuck時，您可以在特定埠(-p)上運行ECMG功能，並帶有詳細選項(-v)和所需的調試級別(-d#)。

範例：

```
sudo tsecmg -p 3337 -v -d7
```

驗證

在cBR-8上

在cBR-8上配置影片會話後，您可以驗證會話是否已建立，因為這是一個基於表的配置，該會話始終存在，並且未顯示任何輸入流：

```
acdc-cbr8-2#show cable video session all
```

Session	Output	Frequency	Streaming	Sess	Session	Source	UDP	Output		
Input	Output	Input	Output	Encrypt	Encrypt	Low PMV	Session	Output		
Id	Port	Hz	Type	Type	Ucast	Dest IP/Mcast	IP (S,G)	Port	Program	
State	State	Bitrate	Bitrate	Type	Status	Lat NUM	Name			
1048576	1	850000000	Remap	UDP	10.10.10.10			65	1	OFF
ON	0	0	DVB	Pending	N	-	vod1.1.0.1.32.65			

啟動影片流後，您可以看到它以明文形式傳送，根據cBR-8上的指令，如果ECMG尚未啟動，則無法清除：

```
acdc-cbr8-2#show cable video sess logical-edge-device id 1
```

Session	Output	Frequency	Streaming	Sess	Session	Source	UDP	Output		
Input	Output	Input	Output	Encrypt	Encrypt	Low PMV	Session	Output		
Id	Port	Hz	Type	Type	Ucast	Dest IP/Mcast	IP (S,G)	Port	Program	
State	State	Bitrate	Bitrate	Type	Status	Lat NUM	Name			
1048576	1	850000000	Remap	UDP	10.10.10.10			65	1	
ACTIVE-PSI	ON	15403951	15164562	DVB	Clear	N	-	vod1.1.0.1.32.65		

當您同時啟動ECMG時，可以看到影片會話現在已加密：

```
acdc-cbr8-2#sh cable video sess logical-edge-device id 1
```

Session	Output	Frequency	Streaming	Sess	Session	Source	UDP	Output		
Input	Output	Input	Output	Encrypt	Encrypt	Low PMV	Session	Output		
Id	Port	Hz	Type	Type	Ucast	Dest IP/Mcast	IP (S,G)	Port	Program	
State	State	Bitrate	Bitrate	Type	Status	Lat NUM	Name			
1048576	1	850000000	Remap	UDP	10.10.10.10			65	1	
ACTIVE-PSI	ON	15353613	15476997	DVB	Encrypted	N	-	vod1.1.0.1.32.65		

已加密會話的詳細資訊：

acdc-cbr8-2#sh cable video sess logical-edge-device id 1 session-id 1048576

Session Name : vod1.1.0.1.32.65
Session Id : 1048576
Creation Time : Thu Dec 6 14:12:54 2018

Output Port : 1
TSID : 42496
ONID : 100
Number of Sources : 1
Destination IP : 10.10.10.10
UDP Port : 65
Config Bitrate : not specified
Jitter : 100 ms
Processing Type : Remap
Stream Rate : VBR
Program Number : 1
Idle Timeout : 2000 msec
Init Timeout : 2000 msec
Off Timeout : 60 sec
Encryption Type : DVB
Encryption Status : Encrypted

Input Session Stats:

=====
State: ACTIVE-PSI, Uptime: 0 days 00:31:33
IP Packets: In 899927, RTP 0, Drop 0
TP Packets: In 6299489, PCR 6408, PSI 4424, Null 0
Unreference 2212, Discontinuity 0
Errors: Sync loss 0, CC error 795, PCR Jump 7,
Underflow 215, Overflow 4, Block 0
Bitrate: Measured 16483732 bps, PCR 17930489 bps

Output Session Stats:

=====
State: ON, Uptime: 0 days 00:31:33
TP Packets: In 6297330, PCR 6395, PSI 4416,
Drop 12801, Forward 6280113, Insert 6029
Errors: Info Overrun 0, Info Error 0, Block 0, Overdue 54210,
Invalid Rate 0, Underflow 0, Overflow 0
Bitrate: Measured 16433824 bps

PAT Info:

=====
Version 26, TSID 8724, len 16, section 0/0
Program 1: PMT 32

Input PMT Info:

=====
Program 1, Version 28, PCR 100, Info len 0
PID 100: Type 27, Info len 6, (lang eng)

Output PMT Info:

=====
Program 1, Version 5, PCR 49, Info len 6, (CA SYS-ID 10101, PID 79)
PID 49: Type 27, Info len 6, (lang eng)

Output PID Map:

=====
PID 32 -> 48
PID 100 -> 49

以及用於顯示ECMG連線狀態的命令：

```
acdc-cbr8-2#show cable video encryption dvb ecmg id 1 connection
```

```
-----  
ECMG ECMG          ECMG   CA Sys   CA Subsys  PID    Lower  Upper  Streams/  Open  
Streams/ Auto Chan  Slot  ECMG          ECMG  
ID  Name          Type   ID          ID          Source  limit  limit  ECMG      ECMG  
ID          Connections Application  
-----  
1    NAGRA_ELK          nagra  0x2775   0x3        auto   48     8190   1         1  
Enabled  RP    1          Tier-Based
```

```
ECMG Connections for ECMG ID = 1
```

```
-----  
Conn Conn    IP          Port   Channel Conn    Open  
-ID  Priority Address      Number ID      Status  Streams  
-----  
1    1         10.48.88.12  3337   1      Open    1  
-----
```

附註：一旦cBR-8接收到ECM，它就儲存在快取中，並且如果與ECMG的連線丟失，則快取的ECM用於加密，直到收到新的連線。

關於ECMG

由於啟用了調試，您可以看到在ECMG和SCS之間交換的所有消息（請參閱初始圖中所示的呼叫流程）：

```
cisco@simulcrypt:~$ sudo tsecmg -p 3337 -v -d7  
debug level set to 7  
* Debug: setting socket reuse address to 1  
* Debug: binding socket to 0.0.0.0:3337  
* Debug: server listen, backlog is 5  
* TCP server listening on 0.0.0.0:3337, using ECMG <=> SCS protocol version 2  
* Debug: server accepting clients  
* Debug: received connection from 88.88.88.89:56102  
* Debug: server accepting clients  
* 88.88.88.89:56102: 2018/12/06 14:38:35: session started  
* Debug: received message from 88.88.88.89:56102  
  channel_setup (ECMG<=>SCS)  
  protocol_version = 0x02  
  message_type = 0x0001  
  ECM_channel_id = 0x0001  
  Super_CAS_id = 0x27750003  
  
* Debug: sending message to 88.88.88.89:56102  
  channel_status (ECMG<=>SCS)  
  protocol_version = 0x02  
  message_type = 0x0003  
  ECM_channel_id = 0x0001  
  section_TSpkt_flag = 1  
  AC_delay_start = 200  
  AC_delay_stop = 200  
  delay_start = 200  
  delay_stop = 200  
  transition_delay_start = -500  
  transition_delay_stop = 0  
  ECM_rep_period = 100
```

```
max_streams = 0
min_CP_duration = 10
lead_CW = 1
CW_per_msg = 2
max_comp_time = 100
```

* Debug: received message from 88.88.88.89:56102

```
stream_setup (ECMG<=>SCS)
protocol_version = 0x02
message_type = 0x0101
ECM_channel_id = 0x0001
ECM_stream_id = 0x0001
ECM_id = 0x0001
nominal_CP_duration = 100
```

* Debug: sending message to 88.88.88.89:56102

```
stream_status (ECMG<=>SCS)
protocol_version = 0x02
message_type = 0x0103
ECM_channel_id = 0x0001
ECM_stream_id = 0x0001
ECM_id = 0x0001
access_criteria_transfer_mode = 0
```

* Debug: received message from 88.88.88.89:56102

```
CW_provision (ECMG<=>SCS)
protocol_version = 0x02
message_type = 0x0201
ECM_channel_id = 0x0001
ECM_stream_id = 0x0001
CP_number = 0
access_criteria (20 bytes) =
    C9 72 BF D7 70 1E 6D 28 06 9A E8 5F 5D 70 1D 63 AC 1A EC 4A
CP = 0
CW (8 bytes) = 4E 0A 45 9D DC 10 4A 36
CP = 1
CW (8 bytes) = AB FF 00 AA 9C 4F 11 FC
```

* Debug: sending message to 88.88.88.89:56102

```
ECM_response (ECMG<=>SCS)
protocol_version = 0x02
message_type = 0x0202
ECM_channel_id = 0x0001
ECM_stream_id = 0x0001
CP_number = 0
ECM_datagram (188 bytes) =
    47 5F FF 10 00 80 70 35 80 AA 03 00 30 00 10 00 08 4E 0A 45 9D DC
    10 4A 36 00 11 00 08 AB FF 00 AA 9C 4F 11 FC 00 12 00 14 C9 72 BF
    D7 70 1E 6D 28 06 9A E8 5F 5D 70 1D 63 AC 1A EC 4A FF FF FF FF FF
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

* Debug: received message from 88.88.88.89:56102

```
channel_test (ECMG<=>SCS)
protocol_version = 0x02
message_type = 0x0002
ECM_channel_id = 0x0001
```

* Debug: sending message to 88.88.88.89:56102

```
channel_status (ECMG<=>SCS)
```

```
protocol_version = 0x02
message_type = 0x0003
ECM_channel_id = 0x0001
section_TSpkt_flag = 1
AC_delay_start = 200
AC_delay_stop = 200
delay_start = 200
delay_stop = 200
transition_delay_start = -500
transition_delay_stop = 0
ECM_rep_period = 100
max_streams = 0
min_CP_duration = 10
lead_CW = 1
CW_per_msg = 2
max_comp_time = 100
```

* Debug: received message from 88.88.88.89:56102

```
stream_test (ECMG<=>SCS)
protocol_version = 0x02
message_type = 0x0102
ECM_channel_id = 0x0001
ECM_stream_id = 0x0001
```

* Debug: sending message to 88.88.88.89:56102

```
stream_status (ECMG<=>SCS)
protocol_version = 0x02
message_type = 0x0103
ECM_channel_id = 0x0001
ECM_stream_id = 0x0001
ECM_id = 0x0001
access_criteria_transfer_mode = 0
```

疑難排解

在cBR-8上，您可以對設定為調試或雜訊級別的對應管理引擎平台跟蹤進行加密問題故障排除（不要忘記在結束時恢復通知級別）：

set platform software trace sup-veman rp active scs debug

cBR-8和ECMG之間的正确消息交換如下所示：

show platform software trace message sup-veman rp active reverse

```
12/07 15:34:43.963 [scs]: [47872]: (debug): ECMG Send channel_setup for channel_id 1
12/07 15:34:43.965 [scs]: [47872]: (debug): ECMG Received channel_status for channel_id 1
12/07 15:34:43.965 [scs]: [47872]: (info): ECMG Channel 0 setup to ip 10.48.88.12 port 3337
12/07 15:34:43.965 [scs]: [47872]: (debug): Open stream 1
12/07 15:34:43.965 [scs]: [47872]: (debug): ECMG Send stream_setup for channel_id 1, stream_id 1
12/07 15:34:43.965 [scs]: [47872]: (debug): ECMG Received stream_status for channel_id 1,
stream_id 1
12/07 15:34:43.965 [scs]: [47872]: (info): ECMG Stream 1 setup to ip 10.48.88.12 port 3337
12/07 15:34:43.965 [scs]: [47872]: (debug): Request ECM for CP 0
```



```
12/07 15:34:43.965 [scs]: [47872]: (debug): ECMG Send CW_provision with 20 AC bytes for
channel_id 1, stream_id 1

12/07 15:34:43.966 [scs]: [47872]: (debug): Received ECM_response for channel_id 1, stream_id 1

12/07 15:34:43.966 [scs]: [47872]: (debug): ECMGp: Forward ECM pkts to SCS

12/07 15:34:43.966 [scs]: [47872]: (debug): Received ECM for CP 0

12/07 15:34:56.015 [scs]: [47872]: (debug): ECMG Send channel_test for channel_id 1

12/07 15:34:56.016 [scs]: [47872]: (debug): ECMG Received channel_status for channel_id 1

12/07 15:35:18.039 [scs]: [47872]: (debug): ECMG Send stream_test for channel_id 1, stream_id 1

12/07 15:35:18.042 [scs]: [47872]: (debug): ECMG Received stream_status for channel_id 1,
stream_id 1
```

相關資訊

- DVB Simulcrypt技術規範，本文最新建立時：[ETSI TS 103 197 V1.5.1\(2008-10\)](#)
- [技術支援與文件 - Cisco Systems](#)