

Konfigurieren der DVB-C-Laborumgebung mit cBR-8, TSDuck und VLC

Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Konfigurieren](#)

[cBR-8-Videositzungen](#)

[Streamer](#)

[ECMG](#)

[Überprüfen](#)

[Auf cBR-8](#)

[zur ECMG](#)

[Fehlerbehebung](#)

[Zugehörige Informationen](#)

Einführung

In diesem Dokument wird beschrieben, wie Sie ein DVB-C-Laborszenario (Digital Video Broadcasting - Cable) mit dem TSDuck-Toolkit, VLC und cBR-8 konfigurieren.

Voraussetzungen

Anforderungen

Cisco empfiehlt, über Kenntnisse in folgenden Bereichen zu verfügen:

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

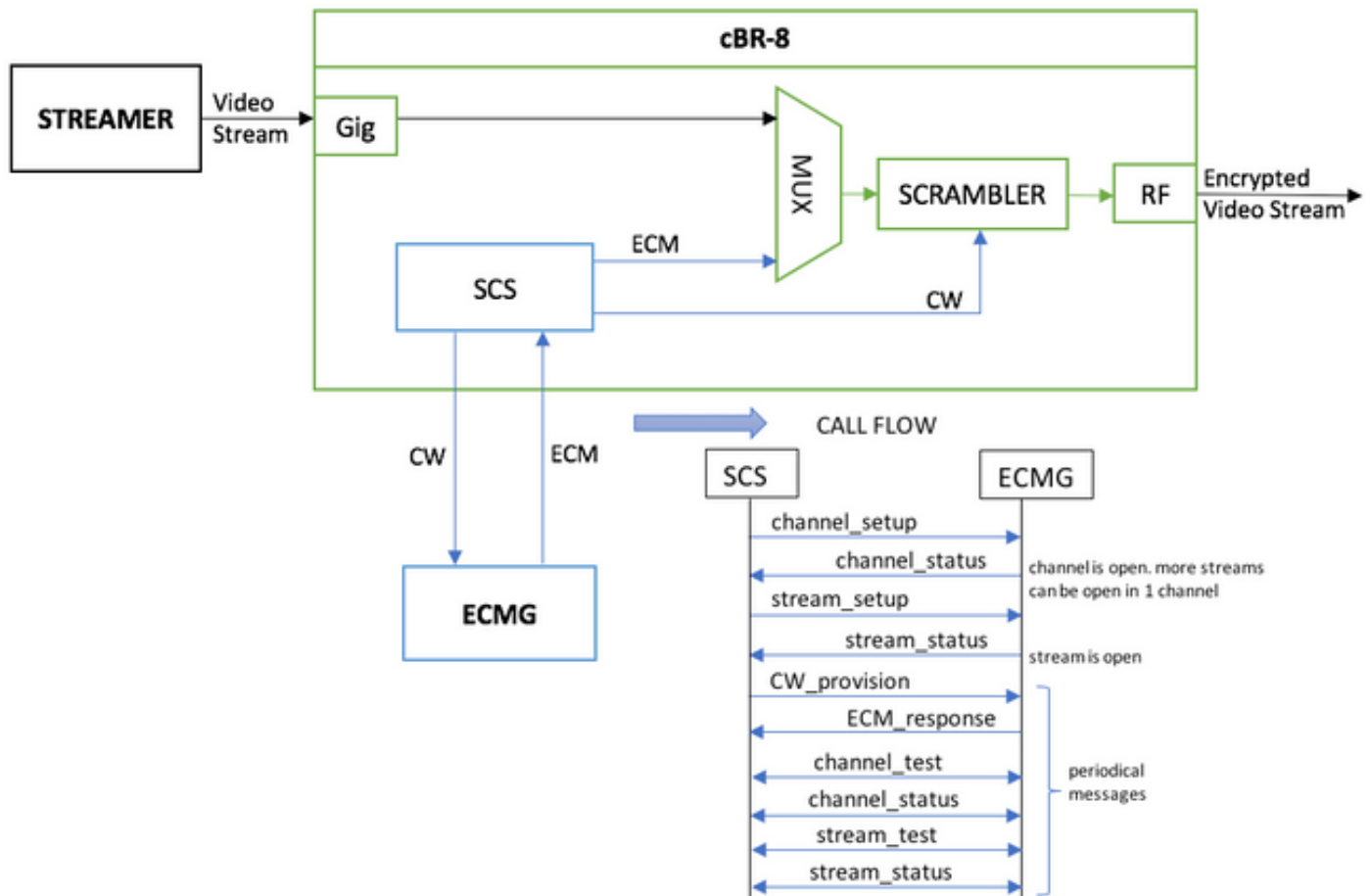
Verwendete Komponenten

Dieses Dokument ist nicht auf bestimmte Software- und Hardwareversionen beschränkt.

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

Hintergrundinformationen

Das in diesem Dokument vorgestellte Szenario, das in der folgenden Abbildung veranschaulicht wird, umfasst cBR-8 als iCMTS, eine als Video-Streamer verwendete Linux Virtual Machine (VM) mit VLC und eine Linux VM mit TSDuck. Das DVB-Symulcrypt-Verschlüsselungssystem wird neu erstellt, wobei der cBR8 als Simulcrypt Synchronizer (SCS) agiert und der TSDuck VM die Rolle des Entitlement Control Message Generator (ECMG) übernimmt, wie es ein Nagra-Server wäre.



Die VM, die als Streamer fungiert, sendet einfach einen lokal gespeicherten Videoclip, der Schleifen durchläuft, um einen kontinuierlichen Stream zu simulieren. Für die cBR-8 ist eine tabellenbasierte (statische) Sitzung für diese Simulation konfiguriert, und es gibt keine Set-Top-Box (STB) oder kein Modem, das den VoD-Stream anfordert, sondern manuell auf dem Streamer initiiert wird.

Wenn der Stream empfangen wird, versucht cBR-8, mit dem konfigurierten ECMG-Server zu kommunizieren, um den Video-Stream zu verschlüsseln und die im Anruffluss in der Abbildung oben beschriebenen Nachrichten auszutauschen. Diese Nachrichten werden klar mit TSDuck ausgetauscht, was gut ist, den Inhalt der Nachrichten und Debug zu analysieren. Auch TSDuck antwortet auf alle gesendeten Anfragen, ohne die Richtigkeit der Parameter wie ca-system-id, access-riteria, etc. zu überprüfen.

Wenn der cBR-8 nicht mit dem ECMG kommuniziert, wird der Stream aufgrund der Anweisung nicht gelöscht.

In einem realen Fall muss den STBs eine EMM (Entitlement Management Message) gesendet werden, die dem Empfänger die Entschlüsselung eines bestimmten Kontrollworts (Control Word, CW) ermöglicht. Die EMMs können über cBR-8 oder über einen separaten Kanal an die

Empfänger gesendet werden, und TSDuck hat auch die Funktion, den EMM Generator (EMMG) zu simulieren.

Konfigurieren

cBR-8-Videositzungen

Hier ein Beispiel zur Konfiguration von DVB-Videositzungen auf cBR-8. Die Zugriffskriterien werden normalerweise vom Zugangsberechtigungssystem (Conditional Access System, CAS) bereitgestellt. In diesem Simulationsfall können Sie sowohl eine zufällige Hex-Nummer als auch die ca-system-ID generieren.

Die virtual-edge-input-ip ist das IP-Ziel des Streams, das in diesem Fall kein echtes Ziel ist, aber es muss die gleiche IP sein, die zum Senden des Video-Streams vom Streamer verwendet wird.

```
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
```

Streamer

Auf diesem Gerät können Sie VLC einfach über die Befehlszeile installieren und einen Stream einer lokal gespeicherten Videodatei starten.

Sie können sich die offizielle [Dokumentation](#) ansehen.

Nach der Installation von VLC zeigt die folgende Befehlszeile, wie Sie einen Stream der Datei cisco-tac-lab.mov starten, die Ziel-IP und den Port, die TSID und den Port auf dem cBR-8 angeben und das Video schleifen, um einen kontinuierlichen Fluss zu simulieren (`—wiederholen`):

```
cvlc cisco-tac-lab.mov —sout
'#doppelte{dst=udp{mux=ts,dst=10.10.10.10:65,tsid=42496,port=65}' —wiederholen &
```

ECMG

TSDuck von der offiziellen Website herunterladen: [TSDuck](#) und weitere Informationen zur Installation und zum Auffinden von Funktionen finden Sie in der Dokumentation des Benutzerhandbuchs.

Wenn TSDuck installiert ist, können Sie die ECMG-Funktion auf einem bestimmten Port (`-p`) mit der ausführlichen Option (`-v`) und der gewünschten Debug-Ebene (`-d#`) ausführen.

Beispiel:

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

Überprüfen

Auf cBR-8

Nachdem Sie die Videositzung auf dem cBR-8 konfiguriert haben, können Sie überprüfen, ob die Sitzung erstellt wurde, da es sich um eine tabellenbasierte Konfiguration handelt, die die Sitzung immer darstellt und keinen Eingabestream anzeigt:

```
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			

Wenn Sie den Video-Stream starten, sehen Sie, dass er laut Anweisung auf dem cBR-8 unverschlüsselt gesendet wird, wenn das ECMG noch nicht aktiviert ist:

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
State	State	Bitrate	Bitrate	Type	Status	Lat NUM	Name	Program

```
-----  
-----  
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
```

Wenn Sie das ECMG auch starten, sehen Sie, dass die Videositzung nun verschlüsselt ist:

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
State	State	Bitrate	Bitrate	Type	Status	Lat NUM	Name	Program

```
-----  
-----  
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
```

Die verschlüsselte Sitzung im Detail:

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

```

Der Befehl zum Anzeigen des ECMG-Verbindungsstatus:

```

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
-----
-----

```

Hinweis: Wenn ein ECM vom cBR-8 empfangen wird, wird es im Cache gespeichert, und wenn die Verbindung mit dem ECMG verloren geht, wird das gecachte ECM für die Verschlüsselung verwendet, bis ein neues System empfangen wird.

zur ECMG

Dank der aktivierten Debugging-Funktion können Sie alle Meldungen sehen, die zwischen ECMG und SCS ausgetauscht werden (siehe Anruffluss, wie in der ersten Abbildung gezeigt):

```

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 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
```

Fehlerbehebung

Auf dem cBR-8 können Sie Verschlüsselungsprobleme beheben, wenn die entsprechenden Supervisor-Plattform-Traces auf "debug" oder "Rauschpegel" eingestellt sind (vergessen Sie nicht, die Benachrichtigungsstufe am Ende wiederherzustellen):

set platform software trace sup-ve-man rp active scs debuggen

Ein korrekter Nachrichtenaustausch zwischen cBR-8 und ECMG sieht wie folgt aus:

```
show platform software trace message sup-ve-man 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
```

Zugehörige Informationen

- Technische Spezifikation DVB Simulcrypt, spätestens bei der Erstellung dieses Artikels: [ETSI TS 103 197 V1.5.1 \(2008-10\)](#)
- [Technischer Support und Dokumentation - Cisco Systems](#)