

Universidad Técnica Federico Santa María Campus Casa Central Valparaíso, Chile



"RETRANSMISIÓN DE LIVESTREAM"

Televisión digital bajo norma ISDB-Tb

DESARROLLADO POR: Sebastián Castillo Araya Correo : sebastian.castilloar@sansano.usm.cl

FECHA: 23 Enero 2023





Índice

1	Introducción	2
2	Requerimientos 2.1 Hardware 2.2 Software	2 2 2
3	Configuración script bash	3
4	Configuración EiTV Playout Professional	4
5	Prueba práctica y resultados5.1Livestream YouTube5.2Livestream Twitch	9 9 11
6	Conclusión	12
7	Referencias	12

Índice de tablas

1	Parámetros comando streamlink-ffmpeg.	•		•	•	 •		•	•	•	 •		•		3

Índice de figuras

1	Interfaz log in
2	Nuevo elemento multimedia
3	Configuración contenido multimedia
4	Nuevo servicio
5	Configuración nuevo servicio
6	Asociación a nuevo servicio
7	Nuevo transport stream
8	Configuración general transport stream
9	Configuración norma ISDB-T
10	Configuración Capa B con su servicio
11	Transport stream con capa A y B configurada
12	Script con livestream de YouTube
13	Ejecución script con livestream de YouTube
14	Transmisión YouTube
15	Script con livestream de Twitch
16	Transmisión Twitch.



1. Introducción

El presente documento tiene como objetivo guíar al lector paso a paso a retransmitir un livestream a la televisión digital, bajo la norma ISDB-Tb, utilizando EiTV Playout Professional. Para eso se presenta al lector, los requisitos que este necesita, preparación de script, preparación de playout y finalmente una prueba práctica. Al seguir correctamente las indicaciones, el lector será capaz de "rescatar" una transmisión en vivo, por ejemplo de YouTube o de Twitch, y podrá retransmitirla a la señal abierta.

2. Requerimientos

Los requisitos de hardware y software indicados a continuación, fueron utilizados en pruebas de transmisión y lograron un resultado satisfactorio.

2.1. Hardware

Para transmitir:

- Computador:
 - CPU: Intel Core i3-5005U 2.00 GHz
 - Memoria Ram: 8GB
 - SSD: 240GB
- EiTV Playout Professional.
- Una antena para transmitir.

Para **recibir**(opcional para observar resultados):

- Televisión.
- Set Top-Box (si la televisión ya cuenta con compatibilidad digital entonces este dispositivo no hace falta).
- Una antena para recibir.

2.2. Software

- Sistema operativo GNU/Linux Ubuntu 22.04 LTS Link oficial.
- ffmpeg 4.4.2 Link oficial.
- Streamlink 3.1.1 Link oficial.





3. Configuración script bash

Para "rescatar" un livestream se utiliza la herramienta de software Streamlink y para transmitir a servidor playout se utiliza la herramienta de software Ffmpeg. Gracias a la compatibilidad existente entre estos programas es que se puede utilizar como fuente en ffmpeg la salida de streamlink. El siguiente comando nos permite realizar dicha acción:

```
$ streamlink [url de livestream] best -0 | ffmpeg -re -i pipe:0 \
-c:v mpeg2video -c:a aac -pix_fmt yuv420p -s 720x480 -aspect 16:9
-streamid 0:180 -streamid 1:181 \
-b:a 128k -b:v 2298k -maxrate 2298k -minrate 2298k -bufsize 2298k \
-f mpegts -muxrate 2700000.0 udp://10.2.51.11:12345;
```

Parámetro	Descripción					
-best	Máxima calidad en fuente livestream.					
-C:V	Codificación de video.					
-c:a	Codificación de audio.					
stroomid	Indica PID de video y audio, por defecto pri-					
-streamu	mero el video luego el audio.					
-b:a	Bitrate de audio.					
-b:v	Bitrate de video.					
-maxrate -minrate	Máximo y mínito bitrate					
-muxrate	Tasa de multiplexación					
-f	Formato de salida					

Tabla 1: Parámetros comando streamlink-ffmpeg.

La tabla 1 muestra el significado de algunas banderas del comando a utilizar, por otro lado respecto al protocolo, ip y puerto indicado al final del comando, este debe corresponder al que se utilizará en servidor playout. Y como información adicional la página oficial de Streamlink indica las páginas con la que es compatible el software, es decir las páginas de donde se puede rescatar las transmisiones.

Ya conocido el comando se recomienda crear un script bash para facilitar las pruebas que se quieran realizar. Para esto se crea con el editor de texto de preferencia el archivo "stream.sh" en el que se escribe lo siguiente:

```
1 #!/bin/bash
2 stream=$1;
3
4 streamlink $stream best -0 | ffmpeg -re -i pipe:0 \
5 -c:v mpeg2video -c:a aac -pix_fmt yuv420p -s 720x480 -aspect 16:9 -
        streamid 0:180 -streamid 1:181 \
6 -b:a 128k -b:v 2298k -maxrate 2298k -minrate 2298k -bufsize 2298k \
7 -f mpegts -muxrate 2700000.0 udp://10.2.51.11:12345;
```

Una vez creado se procede finalmente a dar permisos de ejecución al archivo creado:

\$ chmod +x stream.sh





4. Configuración EiTV Playout Professional

Mediante el navegador de preferencia se accede a la interfaz de configuración que proporciona el playout (http://10.2.51.11), con los siguientes datos de identificación: usuario: eitv y contraseña: playout (figura 1).



Figura 1: Interfaz log in

Ya ingresado a la interfaz, se debe agregar un nuevo elemento multimedia. Para esto se hace click en Add New TS dentro de la sección A/V (figura 2).



Figura 2: Nuevo elemento multimedia

La configuración para el nuevo elemento debe seguir lo dado por la figura 3. La IP debe corresponder a la del playout (a la fecha en que se realiza este informe corresponde a 10.2.51.11) y para el puerto se tiene más libertad, únicamente se debe registrar pues se usará como destino en el script bash. Respecto a los demás datos a rellenar deben corresponder a los que se utilizarán en el script bash.





Name	AV_Webcam_SC
Source	• IP
IP Interface	Address: 10.2.51.11 Gb 1 Port: 12345 Protocol: UDP RTP
PID Video	180
Video Stream Type	[0x02] MPEG-2 ITU-T Rec. H262 video 🗸
PID Audio	181
Audio Stream Type	[0x0F] ISO/IEC 13818-7 AAC Audio (ADTS transport syntax) 🗸
PID PCR	180
TOT time decode	Synchronize system time with TOT from this input
Bitrate Video (kbps)	2298
Bitrate Audio (kbps)	128
Bitrate TS (bps)	2700000

Figura 3: Configuración contenido multimedia

Siguiendo con la configuración, el archivo creado se procede a adjuntar a un Service. Para esto se va al apartado de Services y click en Add New Service (figura 4). Ya creado se configura (figura 5) y luego se asocia el archivo A/V creado (figura 6) al servicio.

05/01/2023 15:45	Services
IP 2: 192.168.0.2	List Registered Service
Services >	Internal Service Name

Figura 4: Nuevo servicio

General	
Name	Service_Webcam_SC
Provider	FFMPEG
Service Type	[0x01] Serviço de televisão digital
EIT Schedule Flag	
EIT Present Following Flag	
H-EIT Flag	
M-EIT Flag	
L-EIT Flag	
Custom	
The following parameters only need to be co defaults.	nfigured if custom values are needed. The value "0" (zero) can be left for
Program PIDs	
Primary Video PID	180 (decimal)
Video Descriptors	(tag:length:byte:tag:length:byte)
Primary Audio PID	181 (decimal)
Audio Descriptors	(tag:length:byte:tag:length:byte)
PCR PID	180 (decimal)

Figura 5: Configuración nuevo servicio





05/01/2023 15:45 IP 1: 10.2.51.11	A/V Playlist for Service 'Service_Webcam_SC'
IP 2: 192.168.0.2	Name A
Services >	Associate Stream
A/V 🎽	Stream AV_Webcam_SC v
Applications	Associate
Stream Events >	Back

Figura 6: Asociación a nuevo servicio

Siguiendo con la configuración de playout ahora se procede a crear un transport stream. Para eso se va al apartado de transport stream y click en Add New TS (figura 7)

05/01/2023 15:34 IP 1: 10.2.51.11 IP 2: 192.168.0.2	Transport Stream
Services >	Please, fill in the Name field.
A/V ⊳ Applications ⊳	Name TS_Webcam_SC Add
Stream Events <a>	
EPG 🎽	
EWBS 🎽	
Transport Stream 🏼	

Figura 7: Nuevo transport stream

Se procede a rellenar los parámetros de acuerdo a nuestro objetivo, indicando interfaz de salida, frecuencia de transmisión, norma a utilizar, modulación entre otros más (Figura 8 y 9).

05/01/2023 15:51	Transmission Configuration
IP 2: 192.168.0.2	Transport Stream: TS_Webcam_SC
Services >	Configuration DVB-C DVB-T DVB-S ISDB-T
A/V Þ	Transmission Mode Terrestrial Delivery System (ISDB-T)
Applications	Out Interface Modulator V
Stream Events <a>	Network ID 1
EPG >	Network Name TS_Webcam_SC
EWBS 屋	Transport Stream ID 1
Transport Stream 🏼	Original Network ID 1
Log >	Broadcast Information Table (BIT)
Configuration >	Affiliation ID 0 1st Descriptors Loop (tag:length:byte:tag:length:byte)

Figura 8: Configuración general transport stream





Transport Stream:	TS	Webcam	SC
	_	-	-

Configuration DV	B-C DVB-T C	VB-S ISDB-T							
ISDB-T Transmission data successfully updated!									
Broadcast type	13-segment TV Broad	cast 🗸							
Frequency	Channel 23 v	527142857 H	Z						
Area Code	17								
Virtual Channel (Remote Control Key ID)	17								
TS Name	EITV Playout								
Transmission Mode	Mode 3: 8k 🗸								
Guard Interval	1/16 🗸								
Partial Reception	Enabled partial rec (implies in Layer A with	eption in the Layer A only 1 segment)							
Emergency	Enabled emergenc	y flag							
Hierarchical Layers		Layer A	Layer B	Layer C					
	Segments	1 ~	12 🗸	0 ~					
	Modulation	QPSK v	16-QAM v	16-QAM v					
	Convolutional Rate	2/3 🗸	5/6 🗸	3/4 🗸					
	Time Interleaving Length	4 🗸	4 🗸	4 🗸					
	Bitrate (bps)	440560	13216800	0					
	Total Bitrate (bps)	13657360							
				Save					

Figura 9: Configuración norma ISDB-T

AVISO: Se configuró el streaming para utilizar la capa B de la norma ISDB-Tb, para que pueda ser reproducido en TV (a través de set top-box), pero aún así es necesario adjuntar un servicio en la Capa A. En la figura 10 se adjunta servicio a capa B, el procedimiento para la capa A es igual. Se aprovechó un servicio ya creado previamente para adjuntar a transport stream (figura 11).

05/01/2023 15:47 IP 1: 10.2.51.11 IP 2: 192.168.0.2	E Services Layers Classification
Services >	No linked services have been found for this TS.
A/V ≥	
Applications	TS Service Association
Stream Events <a>	Service [FFMPEG] Service_Webcam_SC
EPG 🛓	Layer 🔿 A 💿 B 🔾 C
EWBS 🗾	Service ID 32 Suggested value [32]
Transport Stream ∍	Associate

Figura 10: Configuración Capa B con su servicio



Figura 11: Transport stream con capa A y B configurada

Con esto se da por finalizada la configuración de playout, tan solo queda dar play a transport stream y seguir con los siguientes pasos.





5. Prueba práctica y resultados

Se realizan dos pruebas prácticas, una para YouTube y otra para Twitch. La forma en que estas se realizan es haciendo uso del script bash "stream.sh" creado anteriormente. La sintaxis que este utiliza es la siguiente:

\$./stream.sh [url livestream]

5.1. Livestream YouTube

En primer lugar se escoge una transmisión en vivo a gusto de YouTube y luego se escribe como argumento en el script. La figura 12 muestra como se ve el script en consola al momento previo a ejecutar, por otra parte la figura 13 muestra como se ve script al momento de ser ejecutado.

sebita@sebita-HP-Notebook:~/Workspace/practica\$./stream.sh https://www.youtube.com/watch?v=gC4dJGHWwDU

Figura 12: Script con livestream de YouTube.

sebita@sebita-HP-Notebook:~/Workspace/practica\$_/stream.sh_https://www.voutube.com/watch?v=nCdd]GHWwDU
ffmped version 4.4.2-0ubuntu0.22.04.1 Copyright (c) 2000-2021 the FFmped developers
built with acc 11 ((buntu 1) 2.0.19(buntu))
configuration nefixe/usrextra-wersion=Aubuntu0 22 04 1toolchain=bardenedlibdir=/usr/lib/y86 64.libuy-onuincdir=/usr/include/y86 64.libuy-onuarch=amd
64anble.nidishe.strininanble.nullanble.lakmaanble.lihamanble.lihamanble.lihaluranble.lihabuanble
anale increase and a start a star
- enable tubudet - enable tubudet - enable tubute - enable tubudet under tubudet binder bender tubudet - enable tubudet under enable tubudet - enable tubudet
<pre>Second control contro control control con</pre>
Instante -relative instanta (instanta) - relative instanta) - relative instanta (instanta) - relative instanta) - relative instanta (instanta) - relative instanta (instanta) - relative instanta) - relative instanta (instanta) - relative instanta (instanta) - relative instanta) - relative instanta (instanta) - relative instanta
13 - enable luvyA - enable (LuveU) - enable (LuvzU) - enable (LuvzU) - enable LuvzU - enable (LuvzU) - enabl
te-openation-openation-enable-openation-relate-solid - relate-spirityenable-titorsvgenable-titorityenable-titorsvgenable-titorityenable-titorsvgenable-titorityenable-titorsvg
Libavulit 30. /0.100 / 50. /0.100
LibavCoulet 50.154.100 / 50.134.100
Libavioimat 36. /0.100 / 36. /0.100
(Davoevice 56, 15,100 / 58, 15,100
libavriter /.110.100 / /.110.100
LIDSWSCALE 5. 9.100 / 5. 9.100
(105Wresample 3. 9.100 / 3. 9.100
Lippostproc 55. 9.100 / 55. 9.100
[cti]linfo] Found matching plugin youtube for UKL https://www.youtube.com/watch/v=gc4ajoHwwuU
[cli][InTO] Available streams: 144p (worst), 240p, 360p, 480p, 720p (best)
[cli][into] Opening Stream: 720p (nls)
Input #0, mpegts, from 'pipe:0':
Duration: N/A, Start: 63/29.1188000, Ditrate: N/A
Program 1
Stream #0:0(0x101): Audio: aac (LC) ([15][0][0][0] / 0X000F), 44100 Hz, stereo, T(tp, 130 KD/s
Stream #0:1[0x100]: Video: h264 (Main) ([2/][0][0][0] / 0x001B), yuv420p(tv, bt/09, progressive), 1280x/20 [SAR 1:1 DAR 16:9], 30 tpr, 30 tbr, 90k tbn, 60 tbc
Stream mapping:
Stream #0:1 -> #0:0 (h264 (hative) -> mpeg2video (hative))
Stream #0:0 -> #0:1 (aac (native) -> aac (native))
[mpeg2video @ 0x55a3845268c0] Warning vbv delay will be set to 0xFFFF (=VBR) as the specified vbv buffer is too large for the given bitrate!
Output #0, mpegts, to 'udp://10.2.51.11:12345':
Metadata:
encoder : Lavt58.76.100
Stream #0:0: Video: mpeg2video (Main), yuv420p(tv, bt709, progressive), 720x480 [SAR 32:27 DAR 16:9], q=2-31, 2298 kb/s, 30 fps, 90k tbn
Metadata:
encoder : Lavc58.134.100 mpeg2video
Side data:
cpb: bitrate max/min/avg: 2298000/2298000/2298000 buffer size: 2298000 vbv_delay: N/A
Stream #0:1: Audio: aac (LC), 44100 Hz, stereo, fltp, 128 kb/s
Metadata:
encoder : Lavc58.134.100 aac
frame= 924 fps= 30 q=11.4 size= 10139kB time=00:00:30.91 bitrate=2686.9kbits/s speed=0.998x

Figura 13: Ejecución script con livestream de YouTube.





La figura 14 muestra la sintonización de la señal, en el que se logra recibir la transmisión en vivo tal como se asignó previamente. Como referencia también se muestra la fuente original en YouTube.



Figura 14: Transmisión YouTube.





5.2. Livestream Twitch

Para la señal de Twitch los pasos son los mismos, se busca algún canal que esté en vivo y se pasa como argumento al script, tal como se muestra en la figura 15. Ya preparado el script se procede a ejecutar.

sebita@sebita-HP-Notebook:~/Workspace/practica\$./stream.sh https://www.twitch.tv/reventxz

Figura 15: Script con livestream de Twitch.

La figura 16 muestra la sintonización de la señal, en el que se logra recibir la transmisión en vivo tal como se asignó previamente. Como referencia también se muestra la fuente original en Twitch.



Figura 16: Transmisión Twitch.





6. Conclusión

En este documento se abarcaron los pasos para retransmitir una señal en vivo de internet mediante streamlink, ffmpeg y servidor playout a la telivisión digital de manera satisfactoria.

- Se expusieron los requisitos para realizar el experimento.
- Se explicó cómo preparar un script con tal de facilitar el experimento.
- Se explicó cómo configurar el servidor playout.
- Se explicó cómo retransmitir una señal de YouTube y de Twitch.

7. Referencias

- Página oficial Streamlink. Fuente.