



LINUCA – Asociación Usuarios GNU/Linux de Cantabria

Streaming en Linux (AKA MPEG4IP, DSS, MP4WEB howto)

Por César González, [Bolo](http://www.linuca.org/todos.phtml?id_autor=1) (http://www.linuca.org/todos.phtml?id_autor=1)
Creado el 27/04/2003 18:39 y modificado por última vez el 18/06/2003 00:52

En este artículo, convertiremos Linux en un potente servidor de contenidos audiovisuales usando MPEG4IP (Cisco), Darwin Streaming Server (Apple), MP4WEB y el formato MPEG4.

Actualización Añadida la instalación de [Mp4web](#).

Indice

[1- Intro](#)⁽¹⁾

[2- Instalación de MPEG4IP](#)⁽²⁾

[3- Instalación de Darwin Streaming Server](#)⁽³⁾

[4- Streaming MPEG4IP + DSS](#)⁽⁴⁾

[5- Instalación de MP4WEB](#)⁽⁵⁾

[6- Reproductores compatibles](#)⁽⁶⁾

• **1- Intro.**

Este documento describe como instalar la suite de codificación MPEG4IP y DarwinStreaming Server (DSS en adelante). DSS permite servir archivos mov, mp4, mpeg2y mp3 a traves de RTSP (Real Time Streaming Protocol) y junto con MPEG4IP podremos crear contenidos mp4, mp3, aunque la joya de MPEG4IP es mp4live, que mediante un dispositivo de captura de video soportado por video4linux nos permitirá almacenar y servir en directo contenidos a traves de la red.

MPEG4IP implementa el estandar MPEG4 de ISMA y es desarrollado por Cisco Systems. [ISMA](#)⁽⁷⁾ (Internet Streaming Media Alliance) es una alianza formada por [Apple, IBM, Cisco, Envivio, Hitachi, Sony, Sharp, Philips, entre otros](#)⁽⁸⁾, y lo que propone es la adopción del estandar MPEG4, para la transmisión de medios audiovisuales en internet.

Por lo general, los dispositivos o aplicaciones fabricados por las empresas de la lista podrán interactuar unos con otros, gracias a que todos siguen las especificaciones de ISMA.

Hasta aqui todo muy bonito, pero el MPEG4 está sujeto a patente. Las emisoras tienen que pagar por minuto de emisión y los fabricantes de codificadores y decodificadores MPEG4 también tienen que pagar por cada equipo vendido (por eso aunque MPEG4IP se libere bajo licencia MPL la distribución de los binarios no está permitida). Se libran las organizaciones sin animo de lucro que están exentas del pago de royalties por visionado.

Las pruebas para este artículo se han hecho utilizando una Webcam Creative 3 (modprobe ov511).

• **2-Instalación de MPEG4IP.**

La última versión en el momento de escribir estas líneas es la 0.9.8. La podemos descargar desde <http://prdownloads.sourceforge.net/mpeg4ip/mpeg4ip-0.9.8.tar.gz>.

Una vez descargadas las fuentes y descomprimidas en un directorio, compilamos :

```
$ ./bootstrap
$ make
$ su
# make install
```

Para realizar todo el proceso con éxito, básicamente necesitamos gcc, g++, libtool, autoconf, automake y libgtk-dev.

• **3- Instalación de Darwin Streaming Server.**

Mp4live soporta IPV6 y multicasting, por lo que podríamos emitir por esta vía (en redes locales o islas en inet que soporten IPV6) sin necesidad de usar DSS. Pero si queremos emitir por internet, deberemos olvidar estas maravillas y usar unicast con IPV4 ;-(, aquí es donde entra Darwin que hace de repetidor, sirviendo el stream de mp4live a tantos clientes como nos permita el ancho de banda (recordar que emitimos en unicast y para sacar el caudal total se multiplica el consumo del stream por el número de clientes).

La última versión es la 4.1.3, en <http://developer.apple.com/darwin/projects/streaming/> podeis descargarlo, obligan a registrarse aunque este es gratuito. Si no quereis compilar las fuentes, podeis descargar la versión compilada para RedHat 7.x que funciona perfectamente en Debian Woody/Testing/Sid (no he probado en otras).

Una vez en el directorio donde hemos descomprimido el DSS, ejecutamos

```
#!/Install
Installing Darwin Streaming Server

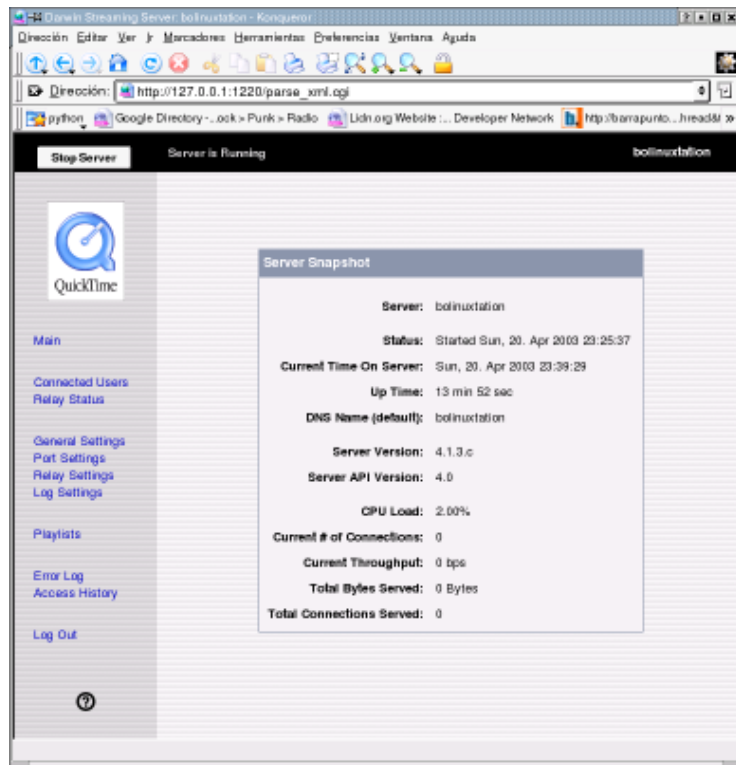
Checking for and Killing currently running Darwin Streaming Server
[....]
Launching streamingadminserver.pl

Installation Complete

Darwin Streaming Server Setup

DSS Administrator Username cannot contain spaces, or quotes, either single or
double, and cannot be more than 255 characters longEnter DSS Administrator
Username :
```

Después de haber fijado el usuario y la contraseña del administrador, lanzamos el navegador a <http://127.0.0.1:1220/> (o a la ip donde esteis instalando el DSS). Esa es la dirección para acceder al panel de administración del DSS.

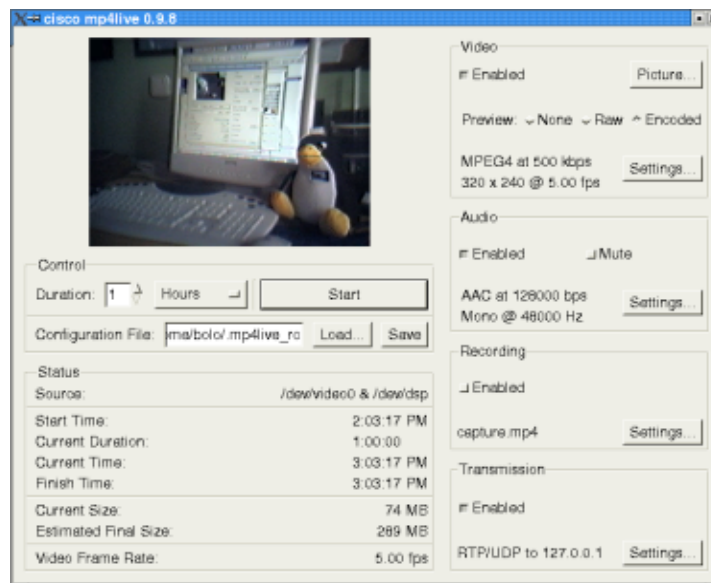


Panel web de administración del servidor destreaming Darwin.

La primera vez que iniciemos sesión se lanzará un asistente que nos guiará en la configuración. Nos pedirá el directorio donde residen los videos que se han de servir y el puerto. Si teneis tambien el apache en la misma máquina no marqueis el streaming a traves del 80. El puerto que se usará en este caso será el 554 TCP y UDP.

• **4- Streaming MPEG4IP + DSS.**

Vamos a probar que la instalación ha ido ok. Lanzamos *mp4live* y veremos algo como esto :



MP4LIVE

Mp4live es la aplicación de la suite MPEG4IP que codifica y sirve la señal audiovisual al DSS. El video es codificado en MPEG4 (ISMA) y en lo referente al audio podemos escoger MP3 o AAC (aunque creo que hay un parche por ahí para añadir soporte para ogg vorbis).

Antes de empezar a emitir, hay que configurar mp4live :

Video Ajustamos los valores (bitrate, ancho, alto, fps, etc) al gusto.

Audio Si queremos que nuestro stream se pueda visualizar por todos los reproductores que soporten el **estandar de ISMA** para MPEG4 (ver apartado Reproductores soportados), deberemos codificar el **audio en AAC**.

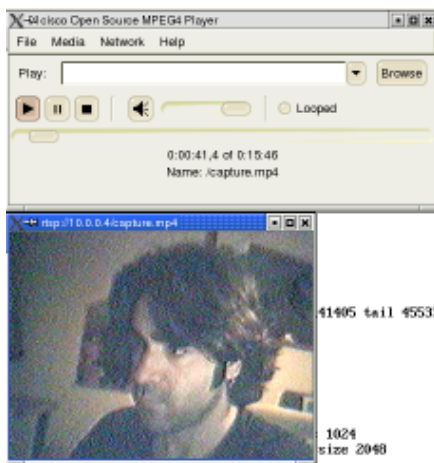
Recording Seleccionamos la ubicación y el nombre del fichero donde se va a almacenar la grabación, lo lógico sería meterlo en algún lugar dentro del "Media Directory" del DSS (por defecto en /usr/local/movies).

Transmission Seleccionamos **Unicast**. En la ip ponemos 127.0.0.1 (en caso de que tengais el DSS y el mp4live en la misma maquina), para los puertos de transmisión de las tramas de video y audio, fijamos dos no privilegiados (>1024) y pares. El fichero **.sdp** es el descriptor contiene información sobre como acceder al stream generado por mp4live en directo. Cuando un cliente quiere acceder al "live stream" debiera solicitar este fichero al DSS. Por ultimo, fijamos un tiempo algo mayor del minuto que viene por defecto y le damos al **Start**.

Ahora viene el momento de comprobar si todo esta tirando como debe :

```
$gmp4player rtsp://127.0.0.1:554/live.sdp
```

Sustituís 127.0.0.1 por la ip del servidor DSS y live.sdp por el nombre del fichero sdp que hayais generado y voilá :



Reproduciendo stream en directo con gmp4player.

Bien, si veis la imagen y escuchais el sonido ya lo teneis hecho ;).

Los videos .mp4 grabados con mp4live, son "servibles" por el DSS una vez almacenados. Para obtener información (bitrate, duración, etc) de un archivo mp4 podemos usar la utilidad mp4info :

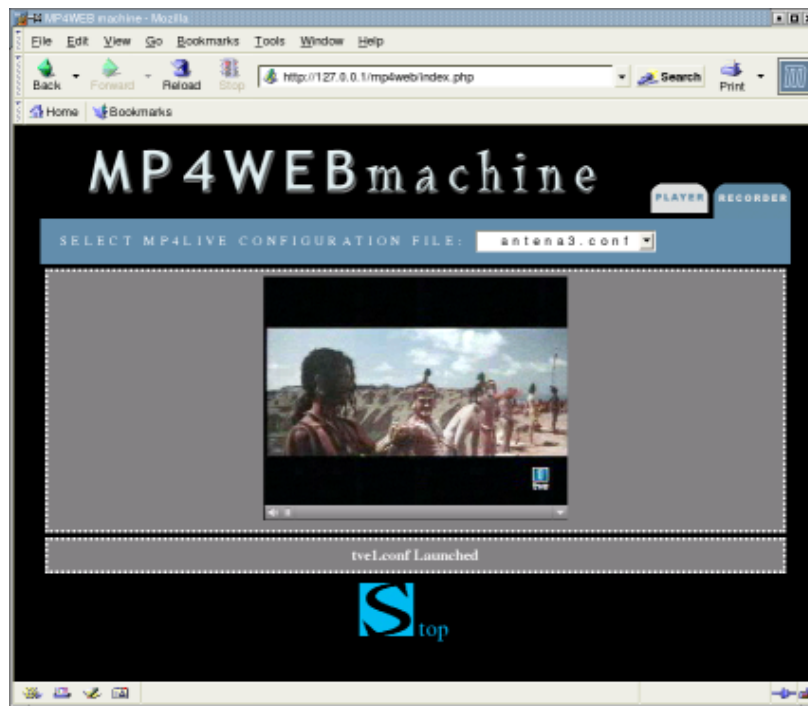
```
$ mp4info /usr/local/movies/prueba1.mp4
mp4info version 0.9.8
/usr/local/movies/prueba1.mp4:
Track  Type      Info
1      video    MPEG-4 Simple @ L3, 219.900 secs, 25 kbps, 128x96 @ 9.99 fps
2      audio    MPEG-2 (MP3), 219.888 secs, 24 kbps, 16000 Hz
3      hint     Payload MP4V-ES for track 1
```

```
4      hint    Payload MPA for track 2
5      od      Object Descriptors
6      scene   BIFS
```

MPEG4IP dispone también de una serie de herramientas (no muy maduras aún) para convertir desde diferentes formatos a mpeg4.

• **5- Instalación de Mp4web.**

Para poder manejar todo remotamente desde un navegador, he hecho un pequeño frontend via web. Con el se pueden lanzar sesiones en vivo, grabarlas, emitirlas, reproducirlas, etc... practicamente todo lo que puedes hacer con mp4live.



Mp4web en acción

[Mp4web](#) necesita apache, php4, php4-pear. En debian, *apt-get install apache php4 php4-pear*. Una vez tenemos esto listo, descomprimos [mp4web-0.1.tar.gz](#) en el arbol web (/var/www en debian).

Ahora ajustamos los permisos del usuario de apache (www-data, httpd, apache, según distribuciones) para que puedan acceder a los dispositivos de audio y video. Añadiendo este usuario al grupo audio y video en el /etc/group:

```
[...]
video:x:44:www-data
audio:x:29:www-data
[...]
```

Tambien tenemos que fijar los permisos del "Media Directory" del DSS (/usr/local/movies) para que el usuario de apache pueda escribir allí:

```
#chown -R www-data:www-data /usr/local/movies
```

Por último, editaremos el fichero configure.php para ajustarlo a nuestras necesidades :

```
//Configuration values
```

```
$darwinServer="rtsp://10.0.2.78:554/";
$dirstreaming="/usr/local/movies/";
$dir_mp4web="/var/www/mp4web/";
```

Introducimos la url y el Media Directory del DSS y el directorio donde hayais instalado mp4web. Apuntais el navegador a <http://vuestra.ip.com/mp4web/> y ya podeis empezar a funcionar.

Mp4web soporta [mplayer-plugin](#) en Linux y en Win32 y MacOS, el Quicktime. El mplayer 0.90 no reprocirá sonido si usamos AAC como codec de audio, si quereis usar AAC necesitareis tirar del repositorio CVS del mplayer (yo tengo el del 16-05-2003 y va muy bien).

- **6- Reproductores compatibles.**

Estos son los reproductores compatibles las especificaciones MPEG4 de ISMA que conozco, aunque hay muchisimos más, se podria afirmar sin equivocarse que cualquier dispositivo o aplicación que fabrique cualquier socio de ISMA será compatible con nuestro stream.

Mp4player (multiplataforma). Es el reproductor que viene con MPEG4IP y desgradaciadamente aún no está en fase estable, y tiene bastantes problemas (sincronizado, perdidas de audio y video, etc).

Mplayer (UNIX). Es el reproductor mas famoso en el mundo libre. Mplayer soporta nativamente la reproducción de MPEG4, pero para que soporte el protocolo RTSP debemos compilarlo linkado a la libreria [LIVE](#)⁽⁹⁾. Instrucciones [aquí](#)⁽¹⁰⁾.

Quicktime (MacOS, Win32). De todos los que he probado este es el que mejores resultados ha dado.

Windows Media Player (Win32). El Windows Media no soporta nativamente ni el protocolo RTSP (Real Time Streaming Protocol) ni el formato MPEG4 de ISMA. Para solucionarlo existe un plugin de la empresa [Envivio](#)⁽¹¹⁾ (miembro de ISMA) que añade soporte RTSP y MPEG4 a WMP.

Real Player (UNIX, Win32). Al igual que en el caso anterior Real Player no soporta MPEG4. Recurriremos otra vez, al plugin de Envivio, si queremos usar Real Player.

Lista de enlaces de este artículo:

1. <http://linuca.org/body.phtml?nIdNoticia=170#pto1>
2. <http://linuca.org/body.phtml?nIdNoticia=170#pto2>
3. <http://linuca.org/body.phtml?nIdNoticia=170#pto3>
4. <http://linuca.org/body.phtml?nIdNoticia=170#pto4>
5. <http://linuca.org/body.phtml?nIdNoticia=170#pto5>
6. <http://linuca.org/body.phtml?nIdNoticia=170#pto6>
7. <http://www.isma.tv/>
8. <http://www.isma.tv/about/ourmembers>
9. <http://www.live.com/>
10. <http://www.live.com/mplayer/>
11. <http://www.envivio.com/>

E-mail del autor: cesar@eureka-sistemas.com

Podrás encontrar este artículo e información adicional en: <http://linuca.org/body.phtml?nIdNoticia=170>