

# ODROID

Magazine

Año Tres  
Num. #32  
Ago 2016

Conoce cómo funciona el

## *Obturador Global*

de la oCam y perfecciona tus habilidades con la cámara



- Penetra en redes WPA usando un ODRROID-XU4

- Crea un clúster XU4 usando una simple topología de proyectos

# Qué defendemos.

Nos esmeramos en presentar una tecnología punta, futura, joven, técnica y para la sociedad de hoy.

Nuestra filosofía se basa en los desarrolladores. Continuamente nos esforzamos por mantener estrechas relaciones con éstos en todo el mundo.

Por eso, siempre podrás confiar en la calidad y experiencia que representa la marca distintiva de nuestros productos.

Simple, moderno y único.  
De modo que tienes a tu alcance lo mejor



## HARDKERNEL



Realizamos envíos de ODROID-C2 and ODROID-XU4 a los países de la UE! Ven y visita nuestra tienda online!

**Dirección:** Max-Pollin-Straße 1  
85104 Pförring Alemania

**Teléfono & Fax**  
telf : +49 (0) 8403 / 920-920  
email : service@pollin.de

**Nuestros productos ODROID se pueden encontrar en:** <http://bit.ly/1tXPXwe>





**E**l periférico **OCAM**, una cámara 720p de alta gama fabricada específicamente para usarla con los dispositivos **ODROID**, pronto tendrá una nueva versión disponible, con la principal novedad de tener incorporado un obturador global en lugar de uno ondulado. Esto te permitirá tomar fotos más profesionales sin deformaciones. El artículo especial de este mes nos muestra la diferencia entre los tipos de obturación que existen para que pueda decidir cuál utilizar en tus proyectos. También disponemos de una nueva fuente de alimentación para el **XU4** que permite enviar más potencia a los periféricos. Está disponible en <http://bit.ly/2aM8yHi>.

**Volumio** también tendrá una nueva versión con muchas mejoras, tenemos la suerte de disponer de un avance de sus nuevas características. **Adrian** continúa su serie sobre la seguridad de las redes inalámbricas con la primera parte de un artículo sobre penetración **WPA**, **Nanik** no habla de la **Capa de Abstracción de Hardware de Android**, **Nicolás** nos hace una introducción a **Recalbox**, el cual te permite convertir tu **ODROID-XU4** en un gran decodificador para tu **TV** y **Michael** nos presenta su fantástico clúster de alto rendimiento. Además, destacamos varios juegos para **Android** y **Linux**, incluyendo el motor **Löve**, que te permite crear tus propios juegos.

ODROID Magazine, que se publica mensualmente en <http://magazine.odroid.com/>, es la fuente de todas las cosas ODROIDianas. • Hard Kernel, Ltd. • 704 Anyang K-Center, Gwanyang, Dongan, Anyang, Gyeonggi, South Korea, 431-815 • fabricantes de la familia ODROID de placas de desarrollo quad-core y la primera arquitectura ARM "big.LITTLE" del mundo basada en una única placa. Para información sobre cómo enviar artículos, contacta con [odroidmagazine@gmail.com](mailto:odroidmagazine@gmail.com), o visita <http://bit.ly/lyplmXs>. Únete a la comunidad ODROID con miembros en más de 135 países en <http://forum.odroid.com/> y explora las nuevas tecnologías que te ofrece Hardkernel en <http://www.hardkernel.com/>



**HARDKERNEL**



Hundreds of products available online for the professional developer and hobbyist alike



**ODROID-XU4**



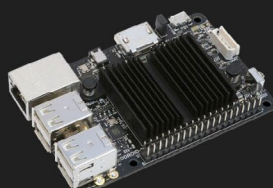
**ODROID-C1+**



**ODROID-C0**



**OWEN ROBOT KIT**



**ODROID-C2**



**VU7 TABLET KIT**



## **Rob Roy, Editor Jefe**

Soy un programador informático que vive y trabaja en San Francisco, CA, en el diseño y desarrollo de aplicaciones web para clientes locales sobre mi cluster ODROID. Mis principales lenguajes son jQuery, angular JS y HTML5/CSS3.

También desarrollo SO precompilados, Kernels personalizados y aplicaciones optimizadas para ODROID basadas en las versiones oficiales de Hardkernel, por los cuales he ganado varios Premios. Utilizo mi ODROIDs para diversos fines, como centro multimedia, servidor web, desarrollo de aplicaciones, estación de trabajo y como plataforma de juegos. Puedes echar un vistazo a mi colección de 100 GB de software ODROID, kernel precompilados e imágenes en <http://bit.ly/1fsaXQs>.

---



## **Bruno Doiche, Editor Artístico Senior**

Nuestro compañero de nutjob tiene más extraña peculiaridad, cada vez que hace zapping por los canales de televisión, si The Matrix está en algún canal, se detiene y lo ve hasta el final. Incluso si se trata de una secuela, no le importa ver una de sus series de película favorita. Teniendo recientemente la suerte de conseguir un fin de semana especial de televisión donde echaron las tres películas y en el canal de dibujos The Animatrix, se lo paso en grande y aunque disfruta mucho, su esposa no entiende lo que le ve a la película y se siente impotente. Así pues llegó la columna de la página 12 que le devolvió la vida.

---



## **Manuel Adamuz, Editor Español**

Tengo 31 años y vivo en Sevilla, España, aunque nací en Granada. Estoy casado con una mujer maravillosa y tengo un hijo. Hace unos años trabajé como técnico informático y programador, pero mi trabajo actual está relacionado con la gestión de calidad y las tecnologías de la información: ISO 9001, ISO 27001, ISO 20000 Soy un apasionado de la informática, especialmente de los microordenadores como el ODROID, Raspberry Pi, etc. Me encanta experimentar con estos equipos y traducir ODROID Magazine. Mi esposa dice que estoy loco porque sólo pienso en ODROID. Mi otra afición es la bicicleta de montaña, a veces participo en competiciones semiprofesionales.

---



## **Nicole Scott, Editor Artístico**

Soy una experta en Producción Transmedia y Estrategia Digital especializa en la optimización online y estrategias de marketing, administración de medios sociales y producción multimedia impresa, web, vídeo y cine. Gestiono múltiples cuentas con agencias y productores de cine, desde Analytics y Adwords a la edición de vídeo y maquetación DVD. Tengo un ODROID-U3 que utilizo para ejecutar un servidor web sandbox. Vivo en el área de la Bahía de California, y disfruta haciendo senderismo, acampada y tocando música. Visita mi web <http://www.nicolescott.com>.

---



## **James LeFevour, Editor Artístico**

Soy un especialista en medios digitales que disfruta trabajando como freelance en marketing de redes sociales y administración de sitios web. Cuanto más aprendo sobre las posibilidades de ODROID más me ilusiona probar cosas nuevas con él. Me traslade a San Diego desde el Medio Oeste de los EE.UU. Continuo muy enamorado de muchos de los aspectos que la mayoría de la gente de la Costa Oeste ya da por sentado. Vivo con mi encantadora esposa y nuestro adorable conejo mascota; el cual mantiene mis libros y material informático en constante peligro.

---



## **Andrew Ruggeri, Editor Adjunto**

Soy un ingeniero de sistemas Biomédicos anclado en Nueva Inglaterra que actualmente trabaja en la industria aeroespacial. Un microcontrolador 68HC11 de 8 bits y el código ensamblador son todo lo que me interesa de los sistemas embebidos. Hoy en día, la mayoría de los proyectos en los que trabajo están en lenguajes C y C ++, o en lenguajes de alto nivel como C# y Java. Para muchos proyectos, utilizo placas ODROID, pero aún sigo intentando utilizar los controladores de 8 bits cada vez que puedo (soy un fan de ATMEL). Aparte de la electrónica, soy un amante de la fotografía analógica y desarrollo la película friki con la que disfruto intentando hablar en idiomas extranjeros.

---



## **Venkat Bommakanti, Editor Adjunto**

Soy un apasionado de los ordenadores desde la bahía de San Francisco en California. Procuo incorporar muchos de mis intereses en proyectos con ordenadores de placa reducida, tales como pequeños modificaciones de hardware, carpintería, reutilización de materiales, desarrollo de software y creación de grabaciones musicales de aficionados. Me encanta aprender continuamente cosas nuevas, y trato de compartir mi alegría y entusiasmo con la comunidad.

---



## **Josh Sherman, Editor Adjunto**

Soy de la zona de Nueva York, y ofrezco mi tiempo como escritor y editor para ODROID Magazine. Suelo experimentar con los ordenadores de todas las formas y tamaños: haciendo trizas las tablets, convirtiendo Raspberry Pi en PlayStations y experimentado con los ODROIDs y otros SoCs. Me encanta trabajar con los elementos básicos y así poder aprender más, y disfrutar enseñando a otros escribiendo historias y guías sobre Linux, ARM y otros proyectos experimentales divertidos.



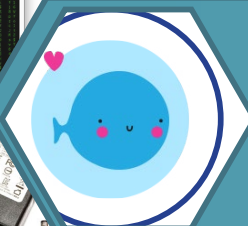
**OBTURADORES DE CAMERA - 6**



**DESARROLLO ANDROID - 10**



**TRUCOS LINUX - 12**



**JUEGOS LINUX: LÖVE (MOTOR) - 13**



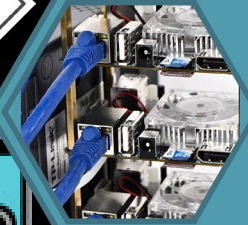
**FUENTE DE ALIMENTACION 5V/6A - 15**



**SEGURIDAD WPA - PARTE I - 16**



**RECALBOX - 20**



**CLUSTER ODROID-XU4 - 22**



**XPOSED FRAMEWORK - 24**



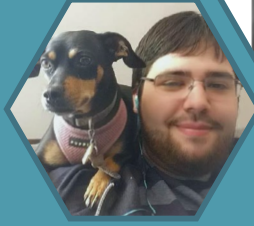
**BROTHERS: A TALE OF TWO SONS - 25**



**VOLUMIO 2.0 - 26**



**THAT LEVEL AGAIN - 27**



**CONOCIENDO A UN ODROIDIAN - 28**

# ¿POR QUE PARECE QUE EL LAPIZ SE DOBLA?

## EL FUNCIONAMIENTO DE LOS DIFERENTES TIPOS DE OBTURADORES DE LAS CAMARAS

por withrobot@withrobot.com

**V**amos a empezar nuestro estudio sobre los obturadores de las cámaras con un sencillo experimento utilizando una webcam y un lápiz. Mantén el lápiz justamente en frente de la webcam con un fondo blanco. Ahora, mueve el lápiz muy rápido de un lado a otro. Como puedes ver en las imágenes capturadas que aparecen más abajo, el lápiz de buenas a primeras parece doblarse por arte de magia.



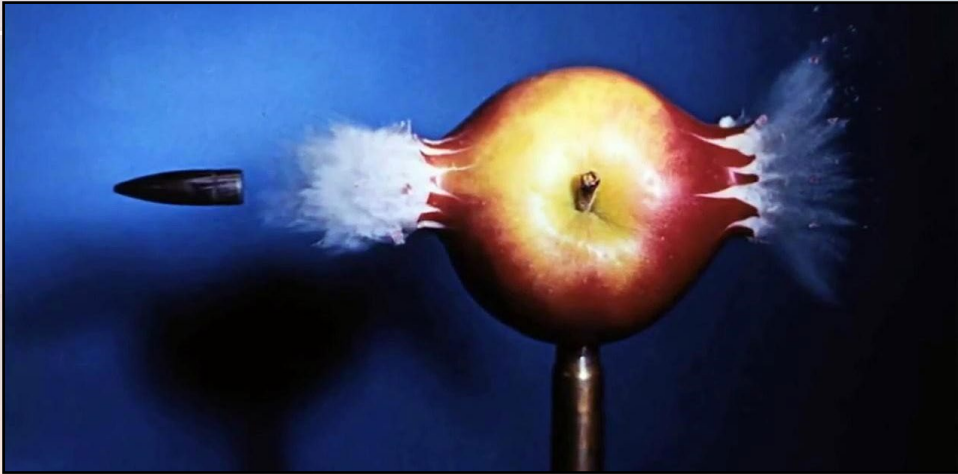
Imagen de un objeto fijo (lápiz) captada por la cámara

¿Qué es lo que hace que el lápiz rígido parezca que se dobla cuando está en movimiento? Para responder a esto, tenemos que analizar el mecanismo obturador de la cámara.

Imágenes de la webcam con un lápiz en movimiento



Puede que hayas visto fotografías muy interesantes tomadas con una cámara de alta velocidad. Un clásico ejemplo de este tipo de fotografías es el de una bala atravesando una manzana.



**Las cámaras de alta velocidad se utilizan para capturar objetos que se mueven muy rápido**

Las cámaras de alta velocidad son especialmente útiles para capturar imágenes de objetos que se mueven muy rápido, como los atletas en los eventos deportivos. Además, estas cámaras son muy apropiadas cuando la cámara en sí es la que se mueve y el objeto a capturar está quieto, como por ejemplo cuando tomamos una imagen desde un dron. Por desgracia, a veces nos encontramos con fenómenos poco agradables cuando trabajamos con imágenes de alta velocidad.

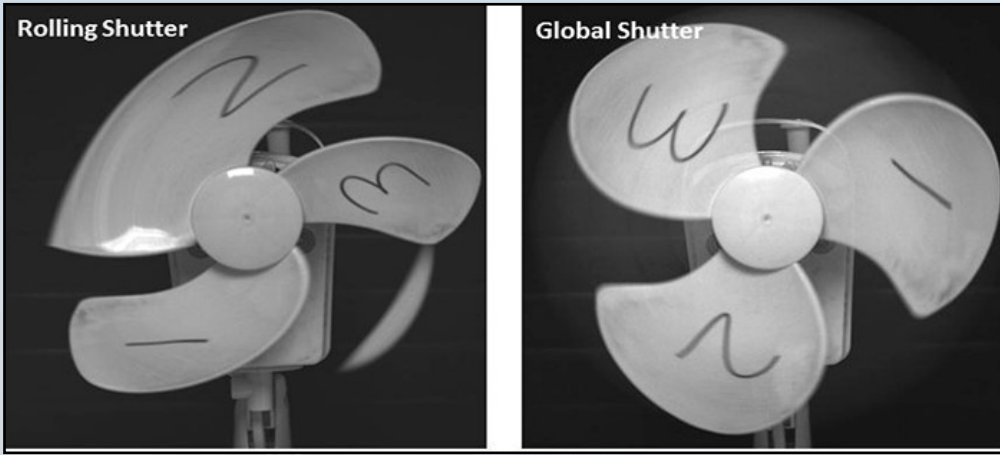


**Típicos problemas de la captura de objetos a alta velocidad. (fuente: Teledyne DALSA)**

La primera imagen “Motion Blur”, se produce cuando un punto específico de un objeto en movimiento aparece en más de un píxel, debido a un excesivo tiempo de exposición. Para reducir el desenfoco del movimiento, tenemos que reducir el tiempo de exposición para que sea tan rápido como sea posible. Sin embargo, un tiempo de exposición más rápido requiere más iluminación.

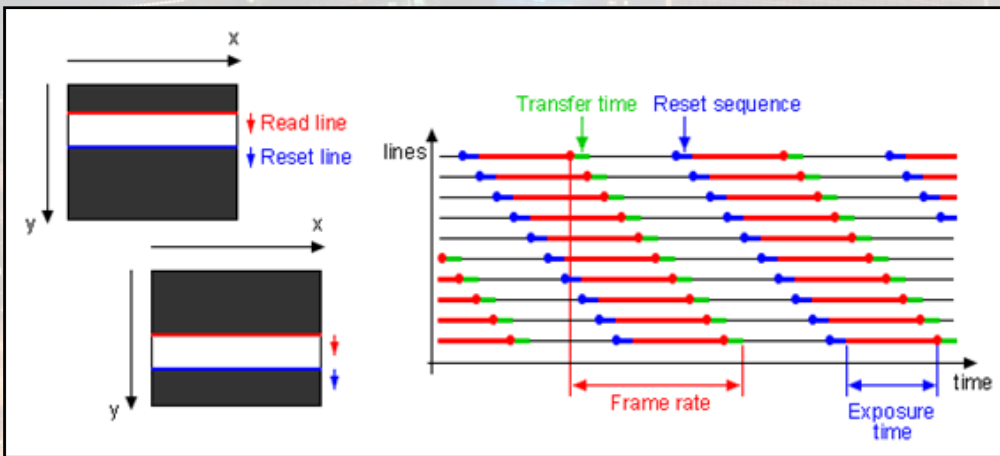
El segundo fenómeno y tema principal de este artículo, es el efecto que produce un obturador ondulado. Hay dos tipos de obturadores en las cámaras digitales; el obturador ondulado y el obturador global. Las imágenes de un ventilador girando en la siguiente página muestran claramente sus diferencias.

El ventilador se deforma en la imagen tomada por una cámara con obturador ondulado mientras que la otra imagen, tomada con una cámara con obturador global, parece normal. ¿Qué causa la diferencia? La respuesta pasa por entender cómo funcionan los dos obturadores.



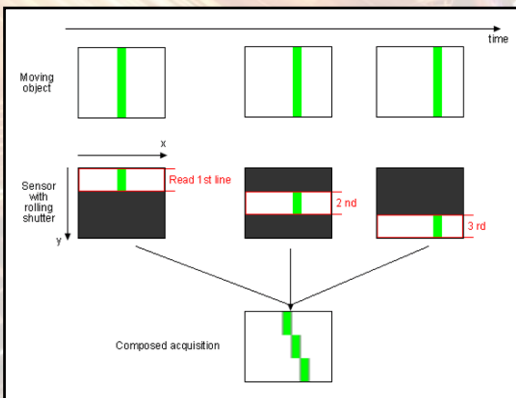
**Imágenes del ventilador en movimiento usando dos tipos de obturadores: Obturador ondulado y obturador global. (Fuente: [www.andor.com](http://www.andor.com))**

Una cámara con un obturador ondulado captura la luz y la envía línea tras línea. El siguiente diagrama puede ayudarte a entender lo que sucede. En este caso, la línea superior de la imagen está expuesta a la luz, lo cual hace que sólo se capte una parte de la imagen. Poco a poco, se van capturando todas las líneas de la imagen hasta formar la imagen completa. Puesto que las líneas se van capturando una tras otra, cada línea es captada un poco más tarde que la anterior.



**La exposición de cada línea empieza y termina en diferentes periodos de tiempo para el obturador ondulado. (Fuente: <https://www.matrix-vision.com>)**

¿Cómo deforma estos diferentes periodos de tiempos de exposición la imagen de un objeto que se mueve muy rápido? Por ejemplo, si tenemos un palo que movemos muy rápido de izquierda a derecha, las posiciones capturadas serán diferentes para cada línea, tal y como se muestra a continuación:



Podemos observar la deformación de un objeto en movimiento más claramente al tomar una foto en un club de golf.

Por el contrario, las cámaras que

**Imagen del obturador ondulado para un objeto horizontal que se mueve muy rápido**







Escena de un club de golf capturada por una cámara con obturador ondulado

usan un obturador global para capturar toda la imagen al mismo tiempo, no van a tener el problema de la deformación. Muy pronto habrá disponible una nueva

cámara para usarla con los ODROIDS. Esta cámara, modelo oCam-1MGN-U, ten-



### Specificaciones

**Sensor:** Sensor de imagen CMOS OnSemi MT9M031

**Lente:** Lente estándar M12

**Tamaño sensor de imagen:** 1/3 pulgadas

**Tamaño del píxel:** 3.75um x 3.75um

**Obturador:** Obturador Global

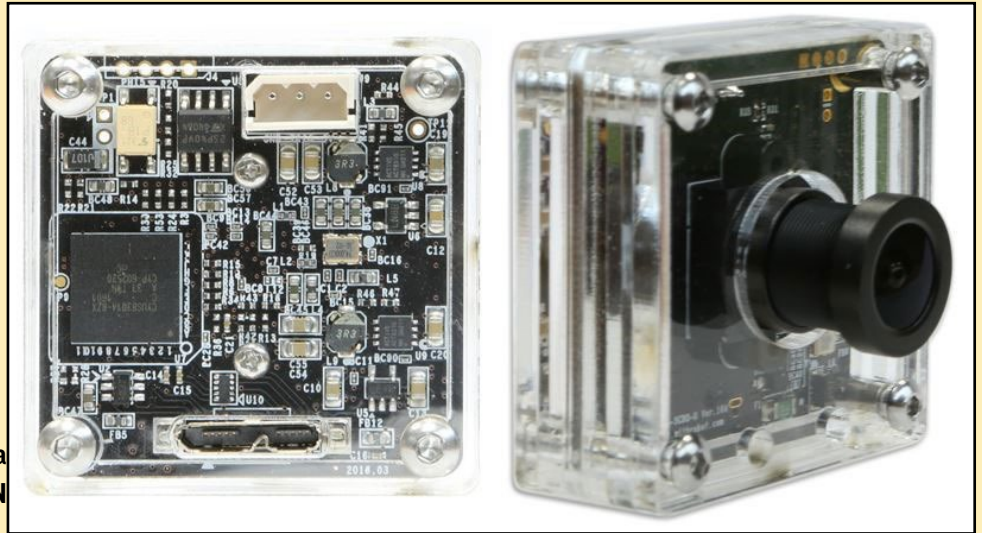
**Interfaz:** USB 3.0 de alta velocidad

**Velocidad de fotogramas:** 1280x960

@ 45fps, 1280x720 @ 60fps, 80fps @

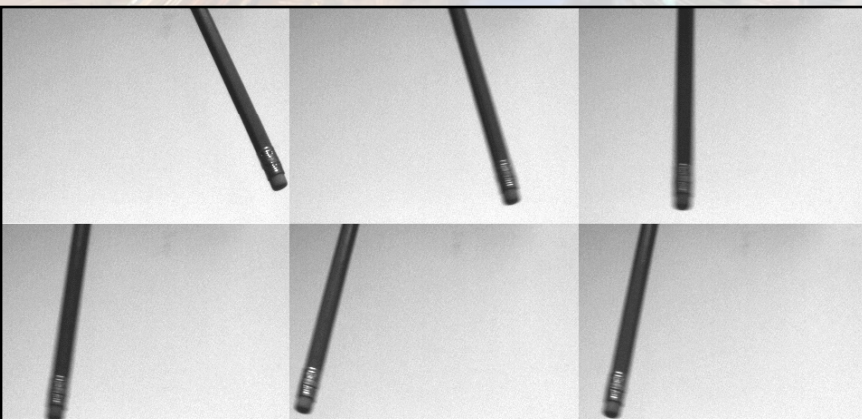
40x480, 320x240 @ 160fps

**Cámara con obturador global para ODROID - OCAM-T-1MGN**



drá el mismo formato que la oCam-5CRO-U.

Para ver las mejoras que aporta esta cámara, hemos capturado la misma imagen del lápiz en movimiento del primer experimento, como se puede apreciar en la siguiente Figura. Claramente, esta cámara ofrece una imagen mejorada sin ningún tipo de deformación. En el próximo artículo, veremos un ejemplo de aplicación muy interesante que utiliza esta nueva cámara con obturador global conectada a un ODROID.



Imágenes de la cámara con obturador global oCam-1MGN-U para un objeto que se mueve muy rápido.



# DESARROLLO ANDROID

## CAPA DE ABSTRACCION DE HARDWARE (HAL)

por Nanik Tolaram



Cuando oímos hablar de nuevos dispositivos Android, siempre nos entusiasamos y queremos ser los primeros en probarlos. Android se utiliza en muchos y muy diversos dispositivos de consumo, desde teléfonos móviles hasta televisores HD. La flexibilidad de Android procede de su pila de software, una arquitectura en capas que permite exportar fácilmente el software a diferentes tipos de hardware y dispositivos. Una de las capas clave que juega un papel fundamental es la capa de extracción de hardware o HAL para abreviar. Disponer de un potente soporte de código para la HAL de Android permite a las placas como ODROIDs ejecutar Android y al mismo tiempo, soportar diferentes tipos de sensores. En este artículo vamos a analizar la capa HAL, centrándonos en la HAL Bluetooth que se utiliza en el ODROID-C2.

### HAL básica

La HAL de Android es la capa de abstracción proporcionada por Google que permite a los proveedores mantener diferentes implementaciones de hardware. La capa HAL es como un traductor entre Android y el hardware, lo cual es necesario para que el entorno de trabajo de Android pueda soportar una amplia gama de configuraciones de hardware. Esto facilita la labor de los desarrolladores a la hora de crear una app que utiliza sensores que funcionan con diferentes dispositivos Android.

Los desarrolladores no necesitan comunicarse con la HAL, ya que de esto se encarga internamente el entorno de trabajo. Existe la posibilidad de que en diferentes hardwares que ejecuta Android, el rendimiento sea distinto utilizando las mismas aplicaciones, esto es algo que los desarrolladores no pueden controlar.

La Figura 1 muestra una estructura

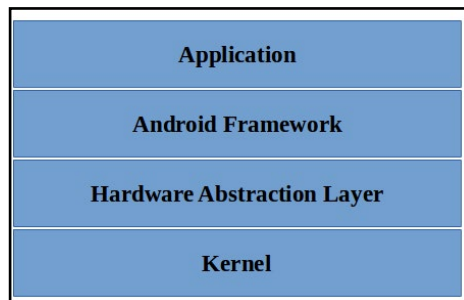


Figura 1 : Pila de Software Android

simplificada de las capas de Android que interactúan con la HAL. La capa “application” es la app de Android que interactúa con la HAL a través de las APIs expuestas en el entorno de trabajo. Esta capa incluye aplicaciones propias del sistema como la app Settings que te permite configurar tu dispositivo, activar la función Bluetooth, ajustar el brillo de la pantalla o modificar los parámetros del WiFi.

### Dentro de la HAL

Como ya he dicho, la HAL es lo que une el entorno de trabajo de Android con el hardware del dispositivo. El código principal de este vínculo lo puedes encontrar en la figura 2. La carpeta “/hardware” incluye varias subcarpetas que

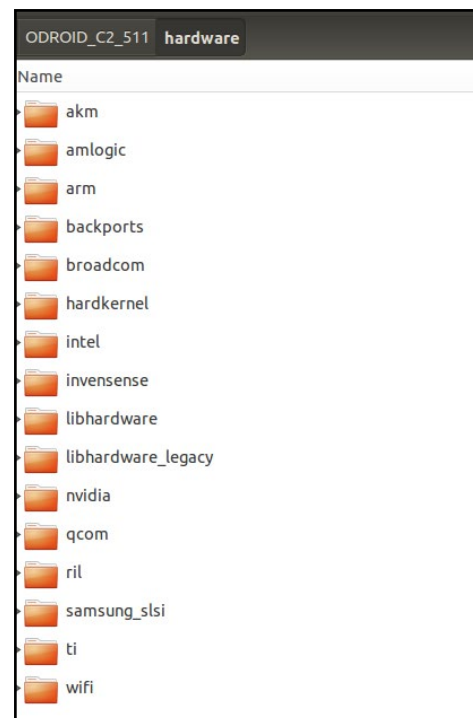


Figura 2: Carpeta hardware

contienen código para diversas implementaciones de hardware y el código de unión definido en Android.

La carpeta que nos interesa es “libhardware”. Como se observa en la Figura 3, existen varios archivos de cabecera en C que corresponden a diferentes vínculos de hardware definidos por Android para diferentes dispositivos, como son el Lector de huellas digitales, el GPS, la cámara, NFC, audio USB y muchos otros. Vamos a analizar el archivo “bluetooth.h” que contiene el vínculo correspondiente a los dispositivos Bluetooth.

Cada módulo HAL utiliza una constante que esta definida como parte del

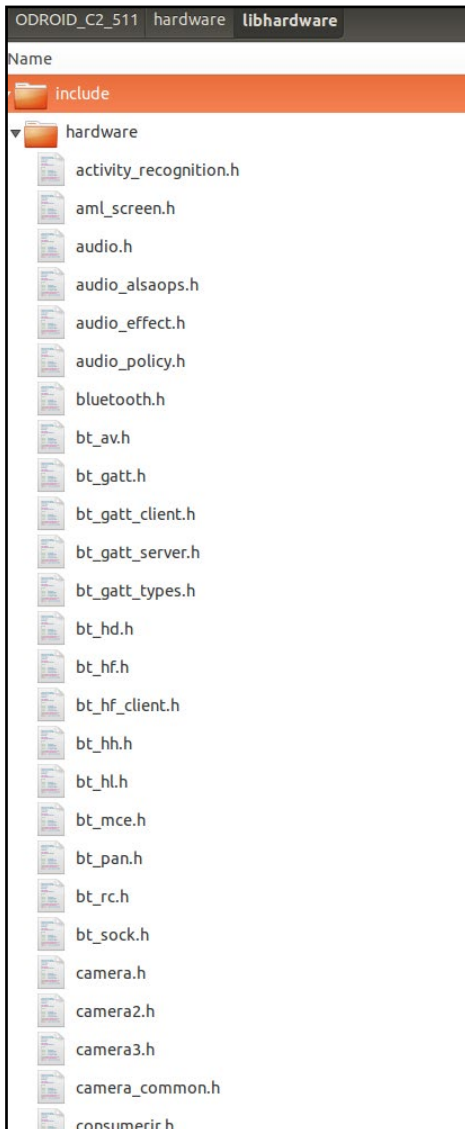
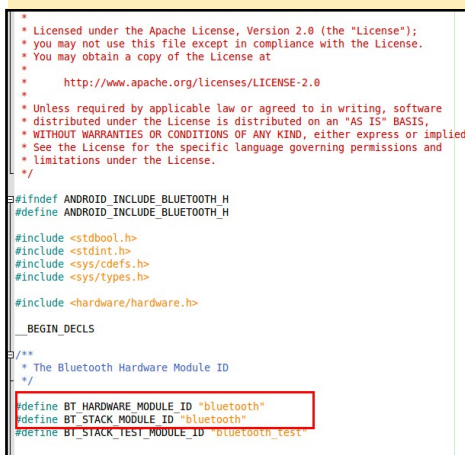


Figura 3: Ficheros cabecera del vínculo HAL

vínculo. Esta constante es utilizada por el entorno de trabajo para cargar la librería correcta del correspondiente hardware. Como vemos en la Figura 4, la constante usada para Bluetooth está definida por una denominada BT\_HARDWARE\_MODULE\_ID.

Figura 4: Fichero cabecera del Bluetooth



El vínculo de la capa HAL de Bluetooth debe implementar la estructura bt\_interface\_t definida dentro del archivo de cabecera bluetooth.h tal y como se muestra en el siguiente trozo de código. En la siguiente sección, verás que HAL Bluetooth aplica esta estructura, junto con la constante definida anteriormente.

```
typedef struct {

// .. break in code .. //

    /** Enable Bluetooth. */
    int (*enable)(void);

    /** Disable Bluetooth. */
    int (*disable)(void);

    /** Closes the interface. */
    void (*cleanup)(void);

    int (*ssp_reply)(const bt_
bdaddr_t *bd_addr,
bt_ssp_variant_t variant,
                        uint8_t ac-
cept,
uint32_t passkey);

    /** Get Bluetooth profile in-
terface */
    const void* (*get_profile_in-
terface)(const char *profile_id);
    int (*dut_mode_configure)
(uint8_t enable);

// .. break in code .. //

    int (*read_energy_info)();
    /** BT stack Test interface
*/
    const void* (*get_testapp_in-
terface)(int test_app_profile);

    /** rssi monitoring */
    bt_status_t (*le_lpp_write_
rssi_threshold)(const bt_bdaddr_t
*remote_bda,
char min, char max);

    bt_status_t (*le_lpp_
read_rssi_threshold)(const bt_
bdaddr_t *remote_bda);
```

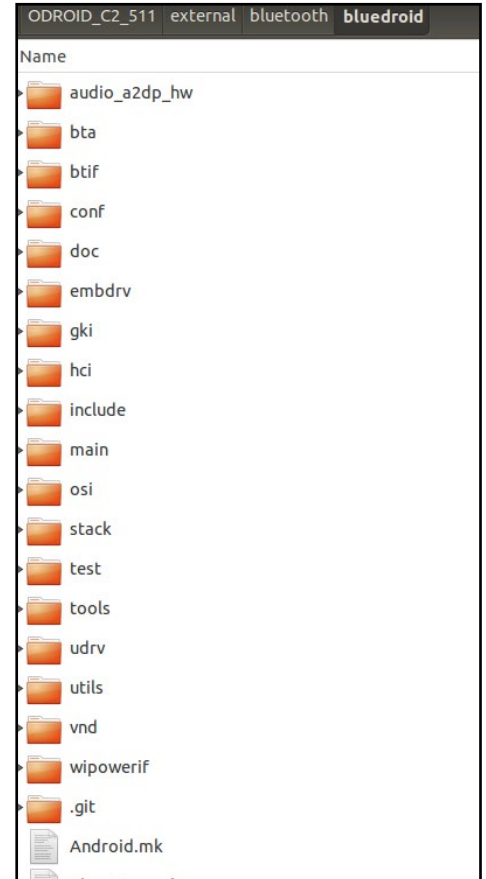


Figura 5: Implementación Bluetooth de Bluedroid

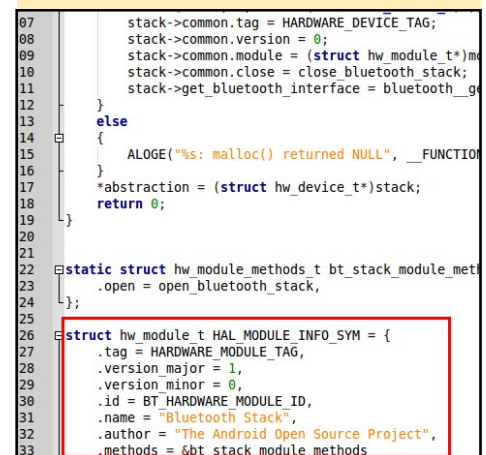
```
// .. break in code .. //

} bt_interface_t;
```

## Implementación Bluedroid

Bluedroid es el proyecto de código abierto que está implementado en el HAL Bluetooth de Android y que reside dentro del directorio “external/bluetooth”, tal y como muestra la Figura 5.

Figura 6: Implementación del HAL Bluetooth Bluedroid



# THE MATRIX

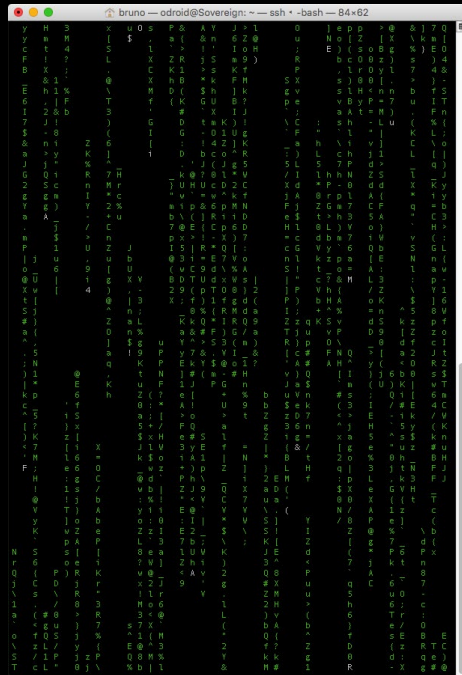
## SIEMPRE SERA INCREIBLE

por Bruno Doiche

El final de una época lejana llamada los años noventa, cuando las bandas de los chicos recorrían el mundo y las chicas llevaban collares y zapatos con plataforma altísimas.

Por aquel entonces, llego a los cines una película diferente e increíble, llamada The Matrix. Junto con ello, todo el mundo empezó a instalar proyectores de pantalla para emular los terminales icónicos que ejecutaran el código de Matrix. Si el friki que llevas dentro quiere volver a verlo, simplemente ejecuta este comando en tu ODROID, te llevará de vuelta a tus años de adolescencia:

```
$ sudo apt-get install cmatrix
```



Nunca dejará de ser buenísimo.

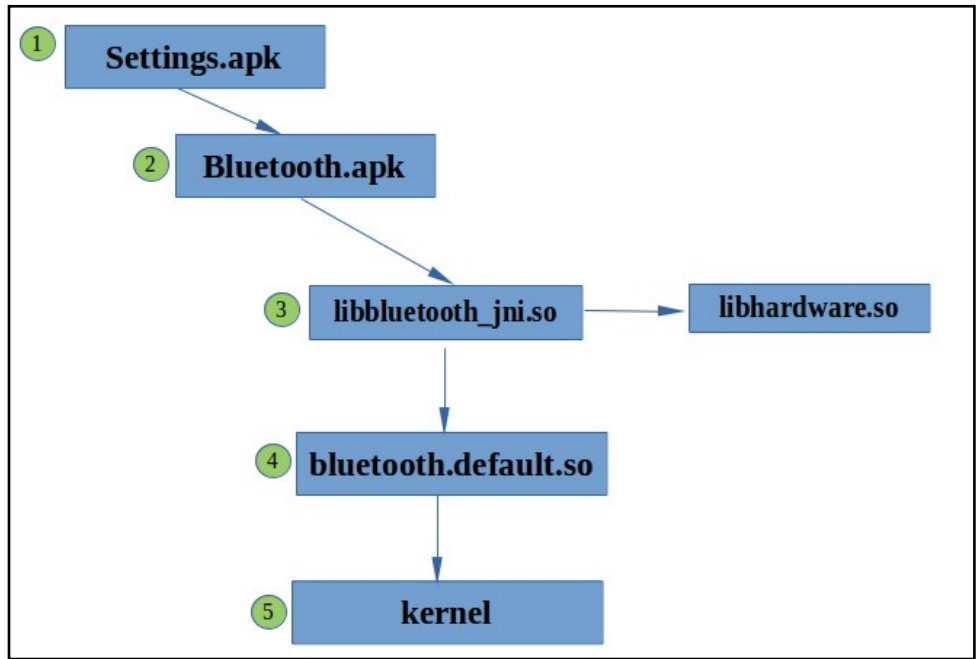


Figura 7: Flujo de interacción del HAL Bluetooth

La implementación del vínculo Bluetooth la puedes localizar dentro del archivo bluetooth.c que se encuentra en la carpeta “bluedroid/btif/src”. Ésta utiliza la estructura hw\_module\_t, definida por HAL\_MODULE\_INFO\_SYM, que es la forma estándar de definir una estructura HAL. Esto permite que el entorno de trabajo encuentre y se vincule fácilmente con la implementación HAL. El campo methods contiene la función de apertura que se activa al iniciar la HAL.

## HAL Bluetooth

Ahora vamos a echar un vistazo a las diferentes capas de aplicación dentro y fuera del entorno de trabajo para mostrar la forma en la que se utiliza la HAL.

Settings.apk es la aplicación Android que normalmente utilizas para encender y apagar el Bluetooth, así como para emparejarte con los dispositivos cercanos. El Bluetooth.apk es el “enlace” que proporciona el servicio para que las apps se comuniquen con la HAL Bluetooth y la pila Bluetooth.

El código interno de Bluetooth.apk es el responsable de cargar la librería compartida libbluetooth\_jni.so con la ayuda de libhardware. De modo que, además de hacer accesibles los servicios, permitiendo que otras aplicaciones Android

puedan interactuar con la pila Bluetooth de bajo nivel, esta aplicación es responsable de activar la librería del entorno de trabajo para cargar la librería Bluedroid, que podemos ver dentro del archivo packages/apps/Bluetooth/jni/com\_android\_bluetooth\_btservice\_AdapterSer-

```

method_deviceFoundCallback = env->GetMethodID(jniCallbackClass, "deviceFoundCallback", "(I)I");
method_pinRequestCallback = env->GetMethodID(jniCallbackClass, "pinRequestCallback", "(I)I");
method_sspRequestCallback = env->GetMethodID(jniCallbackClass, "sspRequestCallback", "(I)I");
method_bondStateChangedCallback = env->GetMethodID(jniCallbackClass, "bondStateChangedCallback", "(I)I");
method_aclStateChangedCallback = env->GetMethodID(jniCallbackClass, "aclStateChangedCallback", "(I)I");
method_getMtuL2cap = env->GetMethodID(clazz, "getMtuL2cap", "(I)I");
method_acquireMtuLock = env->GetMethodID(clazz, "acquireMtuLock", "(Ljava/lang/String;)I");
method_releaseMtuLock = env->GetMethodID(clazz, "releaseMtuLock", "(Ljava/lang/String;)I");
method_deviceInstancesFoundCallback = env->GetMethodID(jniCallbackClass, "deviceInstancesFoundCallback", "(I)I");
method_energyInfo = env->GetMethodID(clazz, "energyInfoCallback", "(I)I");

char value[PROPERTY_VALUE_MAX];
property_get("bluetooth.mtu_stack", value, "");

const char *id = (strcmp(value, "1")? BT_STACK_MODULE_ID : BT_STACK_TEST_MODULE_ID);
err = hw_get_module(id, (hw_module_t const**)(&module));

if (err == 0) {
    hw_device_t *abstraction;
    err = module->methods->open(module, id, &abstraction);
}
  
```

Figura 8: Cargando librería compartida Bluedroid

vice.cpp, tal y como se muestra en la Figura 8.

La pila bluetooth Bluedroid dentro de la librería compartida bluetooth.default.so contiene la implementación de la especificación Bluetooth finalizada, como es la vinculación, la seguridad y mucho más. Esta pila es la única capa responsable de la comunicación con la capa de kernel.



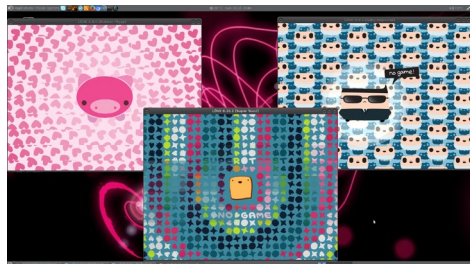
# JUEGOS LINUX

## REPARTE ALGO DE LÖVE (MOTOR)

por Tobias Schaaf



**E**n mi anterior artículo, hablé de un pequeño motor que te permite ejecutar viejos juegos del estilo rol. Este mes, quisiera hablar de un motor diferente llamada Löve, el cual facilita el desarrollo de juegos 2D. El motor Löve (a menudo llamado “Amor”) está basado en scripts Lua que te permite crear fácilmente tus propios juegos.



**Figura 1 – Motor Löve 0.8.0, 0.9.1 y 0.10.1 ejecutándose sobre un ODROID-XU3 sin juegos.**

Para el ODROID, creé 3 versiones de Löve, la 0.8.0 (de Debian Wheezy), la 0.9.1 de Debian Jessie y la 0.10.1 del repositorio git. El motivo de esto es simple. Por desgracia Löve no es tan compatible como debería ser. Un juego hecho para Löve 0.8.0 podría no funcionar en la versión 0.9.1 o posteriores.

De hecho, todas las versiones de Löve hasta la 0.10 requieren OpenGL, lo cual significa que necesita GLshim para ejecutarse sobre el ODROID, y éste está limitado. A partir de la 0.10, Löve soporta OpenGL ES 1 y 2, lo cual mejora la portabilidad de los juegos.

### Instalación

Como siempre, puedes instalar el motor Löve desde mi repositorio usando los siguientes comandos:

```
$ apt-get install love-0.8-odroid
$ apt-get install love-0.9-odroid
$ apt-get install \
love-0.10-odroid
```

Ire añadiendo nuevas versiones a medida que se vayan liberando.

La versión 0.8 y 0.9 de Löve necesita GLshim para funcionar, el cual se instalará como una dependencia. Löve 0.10 se ejecuta usando OpenGL ES 2.

### Juegos

Hay muchos juegos por ahí que utilizan el motor Löve, de hecho, muchos de ellos los puedes buscar directamente en la página principal de Löve.

Wiki: <http://bit.ly/2a3rG0d>

Foro: <http://bit.ly/2atSJHz>

O en otros lugares como indiedb.com: <http://bit.ly/2afQjdr>

Allí, puedes encontrar una gran colección de juegos para Löve, y parece que se van publicando nuevos juegos con bastante frecuencia. Incluso hay algunos juegos de Löve comerciales muy buenos que han sido publicados en Steam y otras plataformas de pago, y otros que están todavía en desarrollo, tales como <http://bit.ly/2afHs8U>. Aunque conseguir que estos juegos de pago funcionen en el ODROID podría ser un poco más

complicado, hay muchos otros juegos a los que se pueden jugar. Me he tomado la molestia de empaquetar algunos de estos juegos y ponerlos en mi repositorio para facilitar su instalación. Puesto que proporciono diferentes versiones de Löve, puedes jugar a otros juegos creados para versiones más antiguas de Löve, junto con los juegos más recientes.

### Mario

Mario0 es un cruce entre Super Mario Bros y Portal, permitiéndote realizar ciertos movimientos locos.

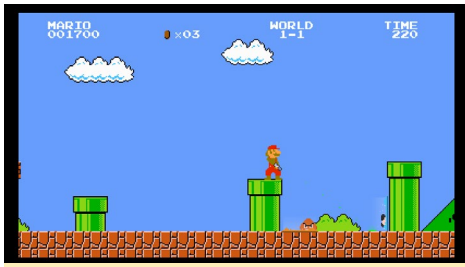
Resulta un poco difícil jugar al principio con un ratón para apuntar en la dirección correcta, aunque usar la pis-



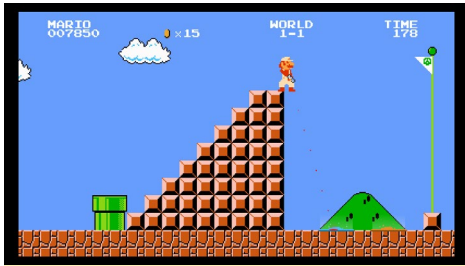
**Figura 2 - Mario0 es un cruce entre Super Mario Bros y Portal, incluso con su propia versión de puzles Portal**

tola portal puede ser muy divertido y te permite realizar algunos movimientos muy buenos. Los desarrolladores, conocidos como “Stabyourself” (<http://bit.ly/2afPLEq>) incluso incluyeron sus propios niveles de Portal, donde tienes que resolver puzles de puertas con diferentes cubos y mediante el uso de portales.

El juego es muy divertido y aunque utiliza el viejo motor Löve 0,8, el cual re-



**Figura 3 - ¡Divertirte con Gumba! ¿Eres capaz de atraparlos a todos?**



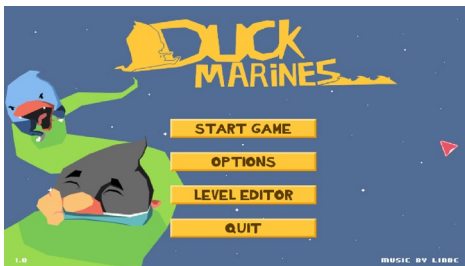
**Figura 4 - Hacer un "salto de altura" nunca fue tan fácil**

quiere OpenGL, funciona bastante bien en ODROIDs utilizando GLshim, y también soporta el modo multijugador. ¿He mencionado que cuenta con su propio editor de niveles?

## Duck Marines

Duck Marines es una reimplementación del juego Rocket ChuChu del Sonic Team, disponible para consolas como la Dreamcast.

El juego es muy divertido, sobre todo porque puedes tener hasta cuatro jugadores en un ODROID. El juego es bastante rápido, a menudo tienes que



**Figura 5 - Duck Marines de tangram es un remake de ChuChu Rocket**

reaccionar muy rápido ante situaciones cambiantes y los minijuegos. Ofrece una gran variedad de mapas diferentes y puedes llegar a crear tus propios niveles, si quieres.

El juego funciona perfectamente en los ODROIDs y utiliza la versión actual



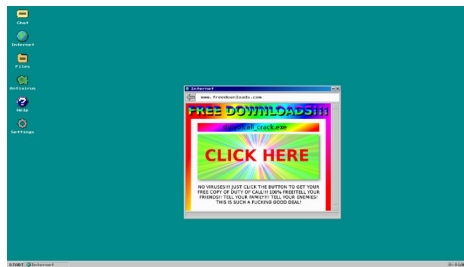
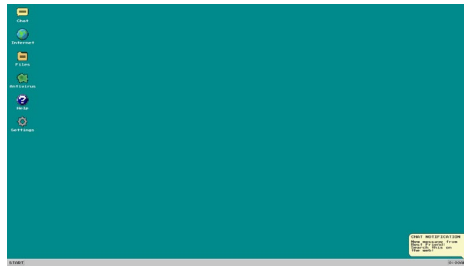
**Figura 6 - Consigue tanto patos como puedas**

0.10 de Löve, lo que significa que se ejecuta sobre OpenGL ES nativo.

## Don't Get a Virus

Este juego es realmente único. Pone de relieve lo que puede suceder si entras en esos sospechosos sitios web de Internet de los que todo el mundo habla.

¿No es como siempre empieza? Antes de que te des cuenta, de repente tu equipo se ve atacado por un virus mortal en una ¡"gigante nave espacial con

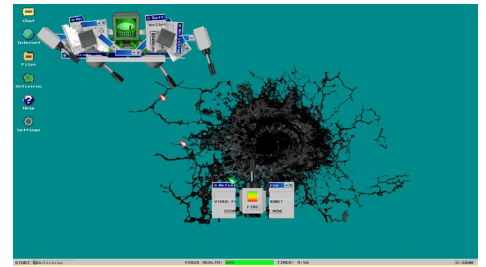


**Figuras 7 y 8 - Así es como comienza todo, hablando con un amigo o visitando un sitio web aparentemente inofensivo**

pistolas láser letales creadas a partir de PCs infectados”!

En fin, se trata simplemente de un juego y afortunadamente estás preparado para defenderte con tu antivirus.

El juego es un pequeño shooter bastante divertido y cuenta con diferentes niveles de dificultad, aunque sólo tiene cinco minutos antes de que el virus se haga con el control de tu equipo.



**Figura 9 - Los virus malignos llegan en gigantes naves espaciales como todos sabemos**

## Mr. Rescue

En Mr. Rescue, eres un bombero que entra en un edificio en llamas para rescatar a las personas de su interior.

Tu única arma es una manguera de incendios con la que puede abrir las puertas y ventanas y hacer retroceder las



**Figura 10 - Mr. Rescue, ¿Tienes lo que se necesita para ser un bombero?**

llamas. Tu única protección es el traje de bombero que te protege del calor, al menos durante un tiempo.

El juego ofrece características muy interesantes, como la línea de visión, con la que tiene una visión limitada,



**Figura 11 - Los civiles de Rescue se arrojan por la ventana antes de quemarse**

dependiendo de dónde te encuentres y la localización de los incendios. Esto hace que sea más difícil encontrar personas para rescatarlas, y nunca sabes lo que te espera detrás de la siguiente puerta o en el siguiente piso.

Tu misión es rescatar a los sobrevivientes y aunque tu instinto te



**Figura I2 - Extinguendo algunas llamas con tu fiel manguera de incendios**

dirá que apagues todos los incendios que ves, no es por lo que estás ahí y perderás si intentan apagarlos todos.

La barra azul de la parte inferior izquierda de la pantalla es la barra del agua que se va reduciendo cuanto utilizas la manguera, pero que también se recuperará lentamente cuando no la usas. Si cierras la manguera antes de que se vacíe, se recuperará y podrás usarla de nuevo en un momento dado, incluso si no está completamente recargada. Sin embargo, si la barra se vacía por completo, tienes que esperar hasta que se recargue completamente antes de poder utilizarla de nuevo.

La barra roja en el centro de la pantalla es la barra más importante. Muestra el calor interior de de tu traje. Si se eleva demasiado, morirás y el juego terminará. Es muy difícil mantener el calor bajo control.

## Reflexiones finales

Löve es un motor muy interesante y el hecho de que esté escrito en Lua facilita a los desarrolladores la creación de juegos. Si estás interesado en desarrollar juegos con Löve, deberías visitar su página web en <http://love2d.org>, donde podrás encontrar diversos tutoriales y consejos, tanto en la wiki como en el foro. También hay tutoriales muy buenos en <http://bit.ly/2afRgCN>.

Signe mi hilo de discusión en el foro sobre el motor Löve dentro de "Games and Emulators" para descubrir mas juegos emocionantes, o ponte en contacto conmigo si has desarrollado tu propio juego y quieren empaquetarlo para los demás.

# FUENTE DE ALIMENTACION 5V/6A DE HARDKERNEL

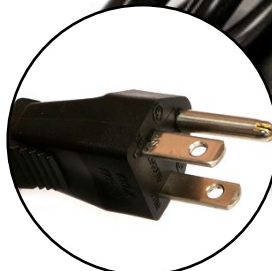
## UNA SILENCIOSA Y ESTABLE FUENTE DE ALIMENTACION DE ALTA CALIDAD PARA EL XU4

por Bruno Doiche

**A**lgunos usuarios de ODROID siempre piensan en hacer cosas nuevas con sus máquinas, inventando constantemente nuevas configuraciones y disfrutando de su portabilidad. Pero hay otros que los utilizan de un modo más simple y práctico, convirtiendo el ODROID en un simple NAS multidisco conectando discos duros en todos los puertos disponibles.

Esto supone una enorme carga para la fuente de alimentación normal de Hardkernel, y cuando todos los discos conectados a los puertos USB se están ejecutando en todo gas, tendemos a obtener errores de E/S que se pueden confundir fácilmente con un disco defectuoso, cuando en realidad no es así.

Prestando atención a nuestras necesidades, Hardkernel nos proporciona una nueva fuente de alimentación 5V/6A que seguirá funcionando perfectamente con tu ODROID-XU4 (y con cualquier disco alimentado por USB).



Con los adaptadores UE - Asia - US, puedes utilizar el cable correcto para el ODROID

# ASUMIR EL RIESGO DE PENETRAR LAS REDES WPA - PARTE I

## PON A PRUEBA TU PROPIO SISTEMA

por Adrian Popa



**E**n nuestros anteriores artículos, atacamos las redes WPS y WEP, pero ahora es el momento de atacar la tecnología más segura que existe en las redes inalámbricas: el cifrado WPA. Como siempre, solicita el consentimiento del propietario de la red antes de intentar penetrar en su red para ahorrarte problemas legales después. ¡Más vale prevenir que curar cuando ponemos a prueba las cosas!

### Fundamentos de WPA

Hay dos tipos de cifrado WPA: El WPA1 que usa TKIP (que es una encriptación WEP más robusto, pero que hoy en día está desfasada y obsoleta) y WPA2 que utiliza AES-CCMP. Nos centraremos principalmente en el WPA2, aunque las técnicas que mostraremos en esta guía se pueden utilizar también con WPA. Desde un punto de vista de la gestión de claves, hay dos tipos de redes WPA:

- WPA-PSK (Pre-Shared Key) - En este escenario, la misma clave de red es conocida por todos los usuarios de la red y cualquier usuario puede ver (descifrar) el tráfico de todos los demás con esta clave. Esta red se utiliza generalmente en las redes domésticas como la tuya en la que probablemente utilizas algún router de serie.
- WPA-Enterprise - Esta red utiliza un servidor RADIUS para autenticar a los clientes, ya sea con nombres de usuario y contraseñas o por medio de certificados individuales en cada dispositivo conectado. La ventaja de esto es que cuando un empleado deja la empresa, su cuenta es desactivada en el servidor en lugar de tener que volver a configurar todos los puntos de acceso y clientes WiFi por seguridad. Una ventaja añadida es que un cliente autenticado no puede esnifar/descifrar el tráfico del resto de clientes, ya que cada cliente tiene una clave diferente, evitando así que un dispositivo sea víctima del posible ataque de otros dispositivos.

El cifrado WPA-Personal puede romperse por fuerza bruta con un handshake cuádruple pero para romper WPA-Enter-

prise, necesitas configurar un punto de acceso falso y configurar un servidor especial RADIUS para obtener las credenciales del usuario (tienes más detalles en <http://bit.ly/2adge8d>). Nos vamos a centrar en WPA-Personal de ahora en adelante.

### El handshake cuádruple

El PSK (Pre-Shared Key) o la combinación de nombre de usuario y contraseña (con redes WPA-Enterprise) en realidad no se utilizan para cifrar los datos en las redes WPA. Sólo se utilizan para autenticar la red. Tras la autenticación, los dispositivos de la red generan claves de cifrado temporales que se utilizan para cifrar los datos en si.

El mecanismo de autenticación WPA conlleva el intercambio de cuatro mensajes EAPOL para configurar las claves de cifrado para la sesión. La información de entrada que tanto el punto de acceso como el dispositivo móvil necesitan en este proceso incluyen:

- Pairwise Master Key (PMK) - Para WPA-Personal éste se calcula concatenando el SSID de la red y tu contraseña en texto plano y ejecutándolo a través de un algoritmo SHA1 4096 veces como esta definido en la función PBKDF2 en RFC2898 <http://www.ietf.org/rfc/rfc2898.txt> .
- ANonce - Un número pseudo-aleatorio de 256 bits generados por el punto de acceso (autenticador)
- SNonce - Un número pseudo-aleatorio de 256 bits generados por el dispositivo móvil (solicitante)
- AA - La dirección MAC del punto de acceso (Dirección del Autenticador)
- SA - La dirección MAC del dispositivo móvil (Dirección del suplicante)

Una vez que el dispositivo finalice el proceso de asociación de red abierta, tanto el punto de acceso (Autenticador) como el dispositivo móvil (Solicitante) generan los Nonces temporales. Estos son el ANonce y el SNonce respectivamente.

Durante el proceso de handshake, el punto de acceso es



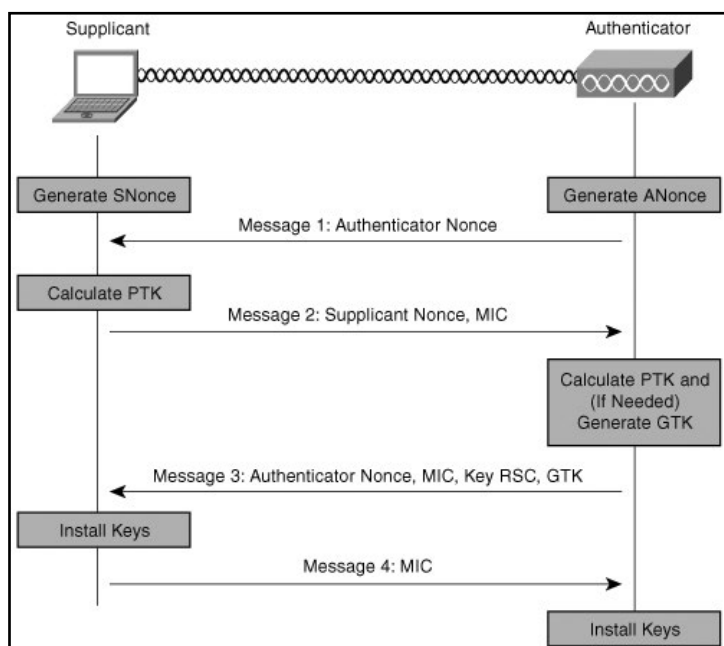


Figura 1 - Diagrama del handshake cuádruple en el cifrado WPA

el que empieza primero enviando un mensaje con el Nonce del Autentificador. El dispositivo móvil utiliza el ANonce y lo combina con la Pairwise Master Key (PMK) conocida, y calcula una Pairwise Transient Key (PTK). La PTK se usa para encriptar los datos de unidifusión y además calcula los Message Integrity Codes (MIC) que tiene una longitud de 256 bits.

El dispositivo móvil entonces envía un mensaje 2 que contiene sus SNonce y el mensaje está protegido con un MIC que se calcula a partir de la PTK derivada de Mensaje 1. El punto de acceso puede calcular sus PTK y comprobar la validez de mensaje 2. Si el MIC es válido, significa que el solicitante conoce la misma contraseña y el proceso continúa.

Después, el punto de acceso genera claves para el tráfico multidifusión que se comparte entre todos los hosts autenticados (GTK - Group Temporal Key), y envía estas claves en el mensaje 3 del proceso de handshake. Ten en cuenta que Mensaje 3 contiene un MIC calculado por el punto de acceso (procedente del SNonce y PMK), que es diferente del valor del MIC calculado por el solicitante. Esto permite que el dispositivo móvil autentique el punto de acceso igualmente. El cliente confirma y envía de vuelta el último mensaje de handshake como una aprobación de este MIC y empieza a utilizar la PTK y GTK para la transferencia de datos. Tienes más detalles sobre el proceso de handshake cuádruple en este video de YouTube <http://bit.ly/2aojWdD>.

## Penetrar un red WPA

Ahora que sabemos cómo funciona la autenticación, vamos a ver lo que tenemos que hacer para romper las redes WPA. En primer lugar, la contraseña de red sólo se utiliza en el proceso de autenticación. Las autenticaciones posteriores generarán diferentes claves temporales que se utilizan para ci-

frar los datos. Por lo tanto, conseguir una de las claves sólo te permitirá descifrar parte de los datos hasta que el cliente vuelva autenticarse o se vuelva a calcular la clave.

El problema es que no puedes obtener realmente la PTK porque no se envía en ningún paquete. Sólo se conoce en los puntos finales. Puedes recalcularla de nuevo para descifrar el tráfico sólo si conoce la información de entrada necesaria para autenticarte en la red. Un atacante que escucha puede conseguir con facilidad la dirección MAC del Autentificador y la dirección MAC del Solicitante ya que se envían claramente, pero es necesario capturar un handshake cuádruple para obtener la ANonce y SNonce. Incluso con esta información, el atacante carece de la PMK, que sólo es conocida por el punto de acceso y los clientes.

El problema se puede reducir a:

- Capturar un completo handshake cuádruple entre un cliente legítimo y el punto de acceso objetivo.
- Usar un método de fuerza bruta para calcular la PMK que combinado con la ANonce puede proporcionar el mismo MIC que el cliente envía en el mensaje 2.

## Capturar el handshake

Para capturar el handshake cuádruple necesitas una interfaz de red en modo monitor y airodump-ng:

```
$ sudo airmon-ng start wlan0
$ sudo iw dev wlan0 del
$ sudo sudo airodump-ng --channel 11 --write nasa
--output-format pcap --bssid BC:EE:7B:8F:6C:B2 --ignore-negative-one mon0
```

El comando airmon-ng pone tu interfaz inalámbrica en modo monitor, mientras que el comando iw elimina la interfaz gestionada (de lo contrario no capturarás mucho tráfico). Airodumpng toma el canal como parámetro, el BSSID del punto de acceso objetivo y escribe una captura de paquetes en formato pcap con el prefijo "nasa" en tu directorio actual. Si no te has dado cuenta a estas alturas, ¡aún estamos intentando piratear la NASA para conseguir nuestra licencia Haxt0r!

Naturalmente, esto significa que tienes que esperar y escuchar hasta que un cliente se conecte, pero si observas que un cliente ya se encuentra conectado, puede ejecutar un ataque desautenticación echándolo fuera de la red y obligándolo a que se autentique de nuevo:

Figura 2 - Airodump-ng es muy bueno para mostrarte la última red que captura un handshake

BSSID	PWR	RXQ	Beacons	#Data	#/s	CH	MB	ENC	CIPHER	AUTH	ESSID
BC:EE:7B:8F:6C:B2	-65	38	408	162	42	11	54e	WPA2	CCMP	PSK	NASA-HQ-WPA
BSSID	STATION	PWR	Rate	Lost	Frames	Probe					
BC:EE:7B:8F:6C:B2	88:30:8A:3F:44:B7	-44	0e-0e	1654	342	NASA-HQ-WPA					

```
$ sudo aireplay-ng -0 3 -a BC:EE:7B:8F:6C:B2 -c
88:30:8A:3F:44:B7 mon0
```

Puede encontrar más detalles en el artículo sobre Inyección Inalámbrica del número de Mayo de 2016 de Odroid Magazine (<http://bit.ly/2azqi7U>).

Aircrack guardará el handshake dentro de un archivo pcap, como el de la Figura 3 (<http://bit.ly/2as3uaL>). Después, podemos utilizar herramientas de penetración como aircrack-ng para buscar la contraseña.

He intentado capturar handshakes con todos los módulos

Time	Source	Destination	Protocol	Info
26.123493	AsustekC_8f:6c:b2	MurataMa_3f:44:b7	EAPOL	Key (Message 1 of 4)
26.139291	MurataMa_3f:44:b7	AsustekC_8f:6c:b2	EAPOL	Key (Message 2 of 4)
26.143883	AsustekC_8f:6c:b2	MurataMa_3f:44:b7	EAPOL	Key (Message 3 of 4)
26.149020	MurataMa_3f:44:b7	AsustekC_8f:6c:b2	EAPOL	Key (Message 4 of 4)

```

Frame 1812: 155 bytes on wire (1240 bits), 155 bytes captured (1240 bits)
IEEE 802.11 QoS Data, Flags: .....T
Logical-Link Control
802.1X Authentication
  Version: 802.1X-2001 (1)
  Type: Key (3)
  Length: 117
  Key Descriptor Type: EAPOL RSN Key (2)
  Key Information: 0x919a
  Key Length: 0
  Replay Counter: 0
  WPA Key Nonce: 2061acd8995130091e0b81b0b54a6fd7f7125f0741e0424a...
  Key IV: 00000000000000000000000000000000
  WPA Key RSC: 0000000000000000
  WPA Key ID: 0000000000000000
  WPA Key MIC: abc841c6b29a3661633ad366f8cc118f
  WPA Key Data Length: 22
  WPA Key Data: 3014010000fac040100000fac040100000fac020000
  
```

Figura 3 - El handshake cuádruple

Wifi de HardKernel, pero no fui capaz de capturar ningún paquete. Probé diferentes posiciones, diferentes kernels, diferentes canales y diferentes dispositivos; llegue a reautenticar mi cliente móvil un centenar de veces, pero la captura no mostró paquetes EAPOL. También lo intente con el WiFi interno de mi portátil (Intel 6205 / iwlwifi), y en este caso sí que fui capaz de capturar el handshake en el primer intento. Esto significa que, o bien hay una limitación de hardware con los módulos WiFi del HardKernel, como por ejemplo la antena, o bien hay algún problema con los drivers que me impide capturar handshakes. Si tú logras capturar algún paquete, por favor deja un comentario en los foros.

## Ataques de fuerza bruta

Aquí tenemos el problema: incluso con un handshake capturado, es necesario echar un vistazo a todas las combinaciones de contraseñas posibles hasta encontrar la correcta, ya que el MIC no es reversible. Tal vez cuando HardKernel lance su placa Quantum mejoren las cosas, ya que esto pone a prueba realmente los límites de la informática convencional.

¿Cuántas combinaciones existen? Momento para las matemáticas: el PSK es una cadena ASCII con una longitud entre 8 y 63 caracteres. Un teclado de Estados Unidos tiene 95 caracteres imprimibles (10 dígitos, 26 letras en minúscula, 26 letra mayúscula y 33 caracteres especiales) así que esto significa que el espacio total para todas las posibles contraseñas PSK es muy grande (véase la Figura 4). Es tan grande que cada átomo del universo conocido tendría 1044 PSK. ¡Con el crecimiento exponencial se pierde el control muy rápido!

$$\sum_{n=8}^{63} 95^n = 39919297033102270412781965613433199719545223215933635382877 \dots 568640629314237842806568803331259120261378523130762379661 \dots 850000000$$

Figura 4 - Número de posibilidades para la PSK

La figura 5 muestra un gráfico con el número de combinaciones para diferentes longitudes de cadena y tipos de caracteres. El número de combinaciones crece exponencialmente con la longitud de la cadena (el eje y de la gráfica es logarítmico, haciendo que el gráfico parezca lineal). Los caracteres dentro de la cadena están representados con una sintaxis normal.

¿Aún no está desanimado? Si conoces el patrón de la PSK

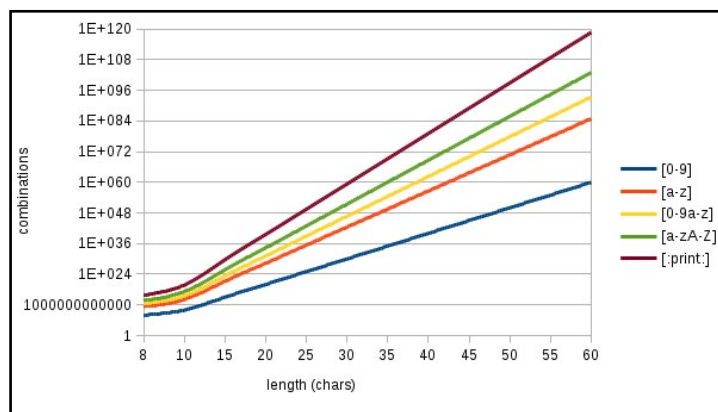


Figura 5 - Número de combinaciones vs longitud de la cadena

o si puedes limitar los caracteres utilizados y su número, como por ejemplo el cumpleaños de alguien en formato ISO 8601, entonces puedes probar el método de fuerza bruta.

Una herramienta diseñada para los ataques de fuerza bruta y que puede beneficiarse de la aceleración GPU Open-CL es oclHashcat. Aunque es una herramienta muy potente y te permite fijar un patrón de contraseña (por ejemplo, letra + letra + mayúscula + número), sólo está escrita para arquitectura Intel, de modo que no podemos utilizarla en nuestros Odroids.

Sin embargo, podemos utilizar “crunch” para generar la lista de combinaciones y aircrack-ng para descodificar vía CPU. Debes tener en cuenta que la lista completa de posibles combinaciones no se almacena porque todas las contraseñas de 8 caracteres en minúscula, por ejemplo, ocupan casi 2 TB de datos. Crunch es muy flexible y te permite especificar la lista de caracteres que deseas utilizar y su posición en la cadena. Visita la página principal SourceForge de Crunch para ver algunos ejemplos.

```
$ sudo apt-get install crunch
```

Si estás usando Ubuntu 14.04, puede instalar Crunch desde GitHub:

```
$ git clone git://git.code.sf.net/p/crunch-wordlist/
```

```
code crunch
$ cd crunch
$ make
$ sudo make install
```

Puedes descargar un archivo de captura desde <http://bit.ly/2as3uaL> para una red con SSID NASA-HQ-WPA que tiene una contraseña de 8 caracteres en minúsculas. Voy a dejarte que encuentres la contraseña por ti mismo, ¡Así que por favor no la reveles en los foros!

Para probar todas las posibilidades con crunch y aircrack-ng puedes usar:

```
$ crunch 8 8 + -t @@@@ @@@@
| aircrack-ng -w - -b
BC:EE:7B:8F:6C:B2 nasa-aaaazzzz-
handshake.pcap
```

La sintaxis indica que Crunch generará cadenas de 8 caracteres con un patrón de letras minúsculas. Esta lista es canalizada en aircrack-ng, que probará todas las combinaciones posibles de la clave de red. Aircrack iniciará tantos subprocesos como pueda y pondrá a prueba tu CPU, así que tu ODROID se calentará bastante.

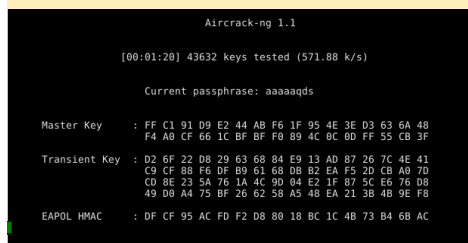
Para usar OpenCL en el descifrado por fuerza bruta, recurriremos a Pyrit:

```
$ sudo apt-get install pyrit
pyrit-opencl
$ pyrit list_cores
$ crunch 8 8 + -t @@@@ @@@@ | py-
rit -r nasa-aaaazzzz-handshake.
pcap -b BC:EE:7B:8F:6C:B2 -i -
attack_passthrough
```

## Rendimiento

Aunque todos los ODROIDs tienen GPUs, el Mali 450 en el ODROID-C1 y ODROID-C2 no soporta Open-CL,

Figura 6 - Crunch + aircrack-ng en acción



así que el ODROID-C1 sólo puede realizar el descifrado por CPU, algo lento (~ 300 PMK / s). El C2 también puede descifrar los hashes con la CPU, llegando a alcanzar una velocidad de ~ 555 PMK/s, aunque también tiene una unidad de cifrado dedicado en la CPU que puede encargarse de algunos cálculos. Pyrit parece usar openssl como backend para hacer el trabajo pesado y OpenSSL debería soportar la unidad de cifrado a través de la extensión del kernel cryptodev. Lamentablemente, ni el clásico kernel ni la versión estándar de OpenSSL vienen con soporte cryptodev, de modo que necesitamos experimentar un poco, no obstante los resultados preliminares parecen prometedores (<http://bit.ly/2adiMTB>).

El ODROID-XU4 tiene una GPU Mali T628, que soporta OpenCL 1.1, de modo que se puede utilizar para descodificar hashes igualmente. Para utilizar OpenCL, necesitas configurar el entorno ejecutando los siguientes comandos:

```
$ sudo mkdir -p /etc/OpenCL/vendors
$ sudo echo /usr/lib/arm-linux-
gnueabi/hf/libOpenCL.so > /etc/
OpenCL/vendors/mali.icd
```

Esto permitirá al XU4 utilizar 2 núcleos Mali en lugar de 2 pequeños núcleos con OpenCL. Ten en cuenta que en relación al funcionamiento, verás que un núcleo Malí tiene el doble de rendimiento que el otro ya que la Mali T628 tiene sus 6 núcleos de procesamiento segmentados en 4/2 en lo que se refiere a OpenCL.

El XU4 usando solo la CPU puede llegar a una velocidad de unos 750 PMK/s limitada especialmente por el calor. Alancé el mejor rendimiento ajustado el regulador en “conservative” y limitando los grandes núcleos a 1,6 GHz. El truco esta en evitar que los grandes núcleos se sobrecalientan, sino los ahogará. Si utilizas la GPU, el rendimiento aumenta hasta unos 1.238 PMK / s - casi

duplicando el rendimiento total. Tal vez con el nuevo sistema de refrigeración azul que vende Hardkernel (<http://bit.ly/2aolJQ2>) con un ventilador Noctua se alcancen mejores resultados.

Otros sistemas con lo que hice pruebas fueron mi PC de trabajo (G3220 Intel @ 2x3.00GHz), mi ordenador portátil (Intel i7-3612QM @ 8x2.10GHz), un equipo para juegos de un amigo (Intel i7-4790k @ 8x4GHz) y un servidor ESX (Intel Xeon X5570 @ 32x3GHz). La Figura 7 muestra los valores de los rendimientos alcanzados. Sin embargo, incluso el programa de decodificación con la mejor CPU se queda perplejo con una GPU de alta gama para juegos (por

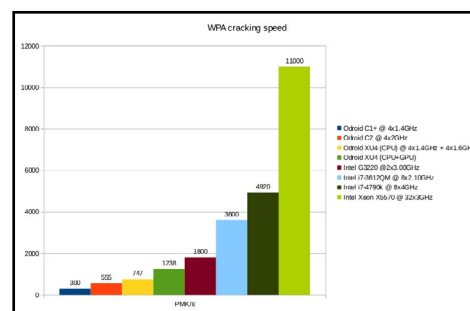


Figura 7 - Rendimiento de la descodificación por dispositivo

ejemplo Nvidia GTX 980) que puede llegar a los 200000 PMK/s. Puedes encontrar los valores previstos para la decodificación por GPU en <http://bit.ly/26v53e7>. Como puedes ver, en una decodificación formal o minería bitcoin, los dispositivos ODROID están fuera de la liga para usarlos en ataques por fuerza bruta, pero pueden ser efectivos en los ataques con diccionario.

Los auténticos atacantes que no temen invertir algo de dinero en el descifrado de contraseñas pueden emplear la ayuda de servidores de Amazon y utilizar la nube de Amazon EC2 para el descifrado por GPU y pagar sólo por el tiempo que lo utilizan. Pueden encontrar más detalles sobre la decodificación industrial en <http://bit.ly/1H4VYtH>. Como siempre, para comentar tus propios ensayos, hacer preguntas o compartir problemas usa el hilo de soporte en <http://bit.ly/2azoM5N>.

# RECALBOX PARA ODROID-XU4

EL DEFINITIVO SISTEMA MULTIMEDIA PARA JUEGOS

por Nicolas Adenis-Lamarre



**R**ecalbox es una liviana distribución Linux que te permite reproducir diversas plataformas y consolas de videojuegos cuando un equipo como el ODROID XU4 lo conectas a un televisor.

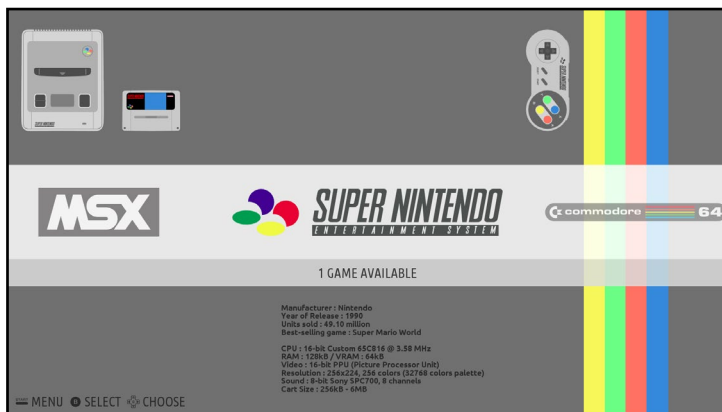


Figura 1 - Pantalla principal de Recalbox

Incluye el software del centro multimedia Kodi y emuladores de consolas como Retroarch, Reicast, Mupen64 e incluso el sistema de juegos por streaming Moonlight. Consulta <http://bit.ly/2a8jAaT> para ver una lista de los sistemas compatibles.

El sistema es plug and play, fácil de usar y por lo tanto muy atractivo para cualquier entusiasta de los juegos. Simplemente conecta tus joysticks compatibles, añade tus propias ROMs y vídeos e invita a tus amigos a unirse. Soporta hasta un máximo de 5 jugadores.

## Características

El centro multimedia Kodi 16.1 ya viene configurado. Puedes controlarlo con tu joystick (bluetooth o USB), el mando de tu televisor o con un dispositivo compatible con la librería de software LIRC. Para empezar, incluye por defecto algunos repositorios Kodi, así como algunas aplicaciones como Youtube y Filmon. Te ayudaran a ver directamente canales como la BBC, tal y como se muestra en la Figura 2.

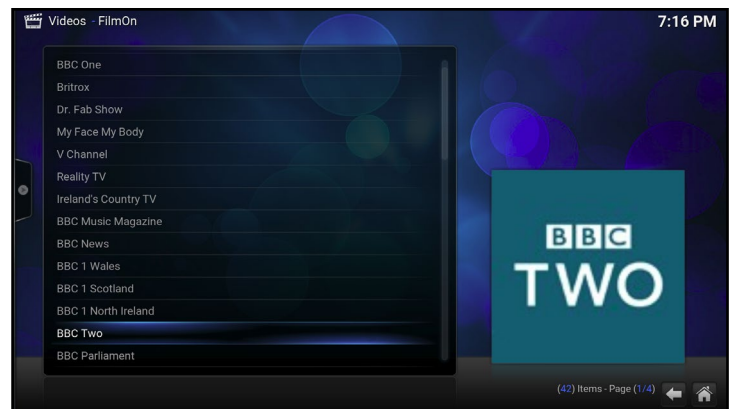
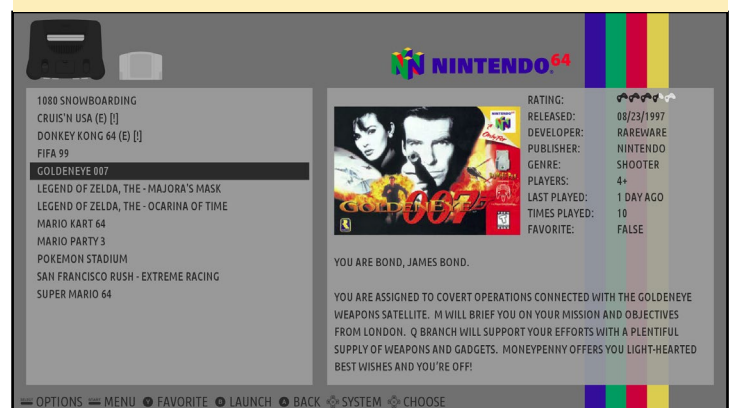


Figura 2 - Complemente Streaming con la BBC

En relación a los juegos, hay unos 50 sistemas disponibles para jugar. El equipo de Hardkernel elegido para Recalbox es el potente ODROID-XU4. Hace que los juegos de la Nintendo 64, Dreamcast y Sony PSP juegos cobren vida, con un rendimiento bastante aceptable.

Los juegos deben copiarse en una tarjeta microSD compatible directamente desde tu ordenador o desde un recurso compartido en red (Samba). También se puede utilizar una memoria USB externa o una unidad de disco duro HDD/SSD. Incluso puedes utilizar un NAS a través de NFS o Samba si deseas utilizar un área de almacenamiento más grande. En el

Figura 3 - Navegando por la biblioteca de juegos de Nintendo



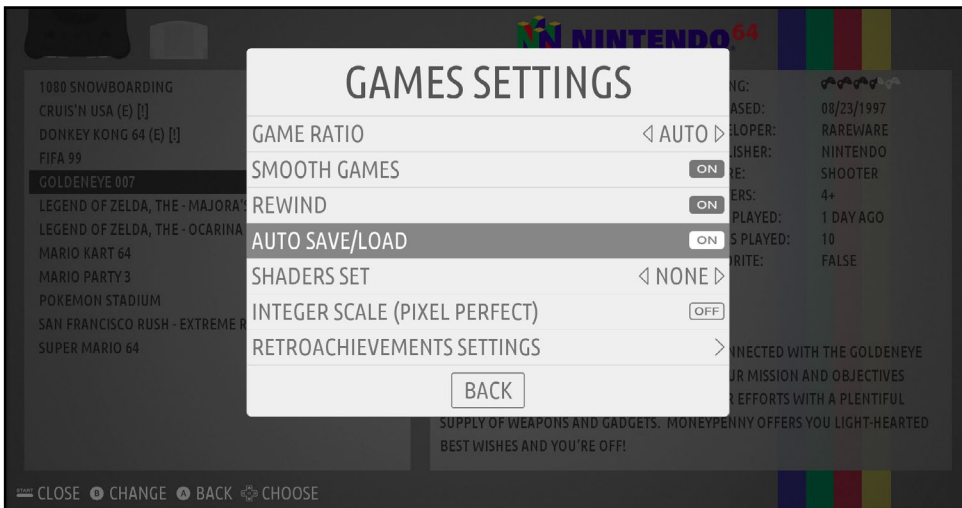


Figura 4 - El menú de configuración de Juegos del Recalbox

futuro, incluso será posible utilizar almacenamiento en la nube como Dropbox o Google Drive para guardar todos o parte de los datos de los juegos. Otras características incluyen shaders, guardado automático, rendimiento retro y sistema de favoritos, por nombrar algunos, como se muestra en la Figura 4.

### Implementación

El Recalbox está compilado con la herramienta Buildroot, lo cual hace posible crear una imagen de Linux partiendo de cero. A diferencia de los sistemas basados en la distribución Linux estándar, este sistema es mucho más sólido. Por ejemplo, el sistema de ficheros root es de sólo lectura. Así que cuando usamos por ejemplo, el sistema con una tarjeta SD, un corte de energía no corromperá el dispositivo. Se trata de un sistema muy liviano, sin servidor Xorg y otras características no esenciales de Linux. Esto permite que su arranque sea más rápido.

La interfaz principal está basada en una versión personalizada de EmulationStation. Ésta permite listar los juegos, los metadatos y las portadas de los juegos.

EmulationStation incluye una interfaz de usuario que permite configurar el idioma, las actualizaciones del sistema, la configuración de red, el rendimiento retro y el joystick. En relación a la emulación, Retroarch facilita la integración de nuevos sistema. Sin embargo, es necesario utilizar el emulador adecuado para asegurarnos un buen rendimiento.

El ODROID-XU4 tiene el botón de encendido, la GPU Mali y los módulos específicos CEC ya implementados. Los componentes más complejos que requieren de un trabajo adicional, son los parches SDL2, la configuración de RetroArch y los parches framebuffer de Kodi.

### Comunidad

La comunidad de RecalBox es muy eficiente y servicial. El soporte está disponible a través de sus foros, IRC, wiki, Facebook y Twitter. Mi agradecimiento a esta comunidad ODROID y a otras como las comunidades Lakka y Build-Root por hacer posible que Recalbox funcione en el ODROID-XU4.

### Referencias

- [www.recalbox.com](http://www.recalbox.com)
- <https://kodi.tv/about/>
- <http://bit.ly/2a8jAaT>



**ODROID Magazine**  
esta en  
**Reddit!**



**ODROID Talk**  
**Subreddit**

<http://www.reddit.com/r/odroid>



# CLUSTER ODROID-XU4

## EL SANTO GRIAL DEL NUCLEO POR VATIOS VS RENDIMIENTO VS PRECIO

por Michael Kamprath

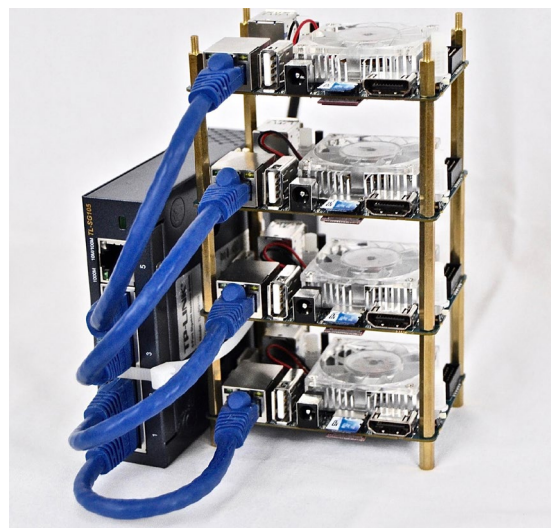
En los últimos años, el tema del big data y de la ciencia de los datos se ha convertido en una tendencia muy importante en un sinnúmero de industrias. Las empresas de alta tecnología de Silicon Valley ya no son los únicos proveedores de temas como Hadoop, la regresión logística y el aprendizaje de máquinas. Estar familiarizado con las tecnologías del big data se está convirtiendo en un requisito cada vez más necesario en los empleos relacionados con la tecnología en cualquier parte. Por desgracia y siendo realista, la experiencia práctica con las tecnologías de big data normalmente implica tener acceso a un costoso clúster de ordenadores para ejecutar tus consultas. Sin embargo, la reciente revolución de los ordenadores de placa reducida ha hecho que la informática distribuida sea accesible a nivel personal y en educación para realizar tareas como esta y muchas otras.

He trabajado con big data durante ocho años. Aunque he tenido acceso a un clúster para procesar petabytes de datos desde hace tiempo, nunca he tenido la oportunidad de diseñar y desarrollar mi propio clúster. Decidí crear un pequeño clúster principalmente para familiarizarme con la instalación básica y el funcionamiento del software de big data y la configuración de un clúster. Mi objetivo consistía en crear un clúster de cuatro nodos por menos de 600\$. También quería que fuese lo suficientemente potente como para ser razonablemente capaz de procesar datos de 10 gigabytes.

Los elementos clave a considerar cuando optamos por la tecnología clúster son el almacenamiento y la E/S de datos, el rendimiento de la red, los núcleos de la CPU y la memoria RAM disponible. Afortunadamente, Hardkernel ha creado un ordenador de placa reducida que supera estas especificaciones: el ODROID-XU4. Con un procesador Samsung Exynos 5422 8-core de 2 GHz, Ethernet Gigabit integrado, varios puertos USB 3.0, 2 GB de RAM, y la disponibilidad del almacenamiento de datos de alto rendimiento con los módulos eMMC y tarjetas microSD UHS-1. El XU4 es un extraordinario ordenador de placa reducida con un coste relativamente bajo.

Con el hardware escogido para los nodos, nuestra primera tarea es diseñar la topología del clúster o cómo conectar los nodos entre sí. Hay varias cuestiones que influyen en esto, sobre todo el tipo de sistema informático distribuido que tengan previsto montar. Los modelos de la informática distribuida, pueden clasificarse a groso modo en big CPU y big data. En este proyecto, nos centraremos en el uso del big data, concretamente en el análisis de los datos. El referente big data más común hoy en día para el análisis de datos es MapReduce, que esta implementado tanto para Apache Hadoop como para Apache Spark, ambas tecnologías de almacenamiento de datos son muy populares y son usadas por muchas de las grandes empresas de tecnología de todo el mundo.

En la mayoría de los clúster MapRe-



duce a escala comercial, la topología de clúster en general tiene cuantos notes quiera conectar un usuario, se utilizan uno o más nodos cabecera para coordinar tanto la actividad informática como el almacenamiento de datos, y un número variado de nodos esclavos para tareas informáticas o de almacenamiento de datos o ambos (ver Figura 1). Piensa en ello como intentar dividir un gran proyecto entre varias personas para mejorar el rendimiento de todos: un director envía la solicitud de proyecto (nodo superior) con varios gestores para coordinar lo que hay que hacer (nodos cabecera) y los empleados que asumen esas tareas y combinan su trabajo (nodos esclavos).

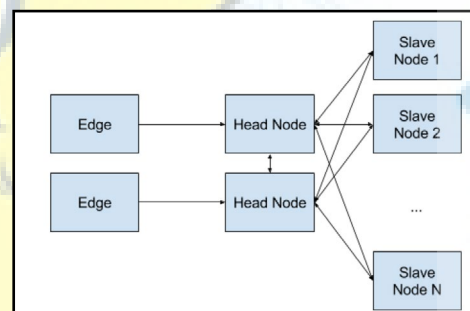


Figura 1 – Típica topología de clúster MapReduce

En nuestro clúster XU4, vamos a combinar el concepto de nodo superior y de nodo cabecera en un único nodo maestro, y luego conectar los esclavos al nodo maestro. Esto significa que el nodo maestro será el nodo en el que los usuarios iniciaran sesión para usar el clúster y el nodo que coordinará los esclavos. Esto también implica que la comunicación de nodo a nodo del clúster tendrá lugar en

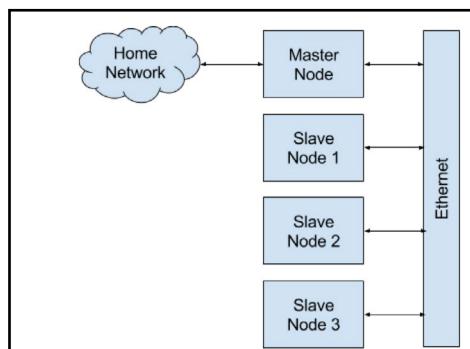


Figura 2 – Topología del Clúster XU4

una red privada, mientras que el nodo maestro necesita tener conexión a la red exterior. Dicho esto, el diseño de la red del XU4 para un clúster de cuatro nodos debería parecerse a la Figura 2.

Esta topología necesita que el nodo maestro se conecte a dos redes diferentes. Sin embargo, el XU4 sólo tiene un puerto Ethernet. Es necesario añadir una segunda conexión de red al nodo maestro con un dongle Ethernet USB 3.0.

El XU4 ofrece dos opciones de almacenamiento: una unidad eMMC y una tarjeta microSD. Ambas tienen pros y contras. La unidad eMMC es extremadamente rápida, mientras que el coste por gigabyte de la tarjeta microSD es más asequible, pero es más lenta que la unidad eMMC. La buena noticia es que el rendimiento de lectura y escritura de una tarjeta microSD UHS-1 está a la par con los discos duros giratorios, que se utilizan normalmente en grandes clúster comerciales. Esto hace que la tarjeta microSD sea una buena opción para el almacenamiento de datos en masa. Sin embargo, la velocidad de la unidad eMMC hace que sea muy atractiva para usarse como unidad de arranque desde la cual se ejecute el software. Teniendo en cuenta que, cada nodo de nuestro clúster tendrá tanto una unidad eMMC para el arranque como una tarjeta microSD para el almacenamiento de los datos en masa. Recomiendo al menos una unidad eMMC de 16 GB para el nodo maestro, ya que será donde tú como usuario, tendrás que trabajar, mientras que puedes ahorrar dinero en las unidades eMMC de 8GB más baratas para los nodos es-

clavos. Para el almacenamiento de datos, busca tarjetas micro SD de 64GB o mayores para cada uno de los nodos.

El conjunto final de materiales necesarios para el proyecto incluye un pequeño interruptor Ethernet para la red interna del clúster, varios cables Ethernet de 6 pulgadas y separadores de PCB para apilar todos los XU4s. También me hice con un módulo UART Serial para el XU4 en caso de necesitar conectarme a un dispositivo directamente para solucionar algún problema, aunque nunca llegue a utilizarlo. Uno elemento que no llegue adquirir y que sería bueno tener a posteriori sería una única fuente de alimentación que pudiera proporcionar la energía de 5V a 4 amperios simultáneamente a todos los nodos, en lugar de disponer de una colección desordenada e ineficiente de adaptadores conectado a una regleta de alimentación. Esto será una futura mejora del proyecto.

Una vez que disponemos de todos los materiales y el clúster está montado, nuestra primera tarea consiste en configurar el sistema operativo y la conexión de red en todos los nodos. Opte por utilizar la actual distribución Ubuntu 15.10 de ODROID para la XU4. Grabé este sistema operativo en cada uno de los módulos eMMC, luego arranque uno por uno cada dispositivo sin la tarjeta microSD (será utilizada más tarde para el almacenamiento) y al mismo tiempo los conectaba directamente a la red doméstica. Esto me permitía conectarme directamente al dispositivo vía SSH tras el primer arranque. Una vez arrancado el dispositivo, busque la dirección IP de cada XU4 recogida por el servidor DHCP de mi casa e inicié sesión. La cuenta de usuario por defecto es “odroid” con la contraseña “odroid”. Una vez conectado, instalé la herramienta ODROID utility para seguir configurando el sistema operativo. Esto se puede hacer descargando directamente la utilidad desde Github:

```
$ sudo -s
$ wget -O /usr/local/bin/odroid-
```

```
utility.sh \
  https://raw.githubusercontent.com/\
  mdrjr/odroid-utility/master/\
  odroid-utility.sh
$ chmod +x /usr/local/bin/odroid-
utility.sh
$ odroid-utility.sh
```

Las tres tareas para las cuales utilizaremos ODROID Utility son: renombrar el nodo, desactivar Xorg y maximizar el tamaño de la partición de la unidad eMMC. Asigne el nombre de Maestro al nodo maestro y a los otros tres: esclavo1, esclavo2, y esclavo3

El nodo maestro tiene que ser además configurado para usar el dongle Ethernet USB 3, para la red externa. Para configurar el nodo maestro y disponer de una conexión a Internet externa mediante el dongle USB, necesitarás crear un archivo llamado “eth1” en el directorio /etc/network/interfaces.d/ con el siguiente contenido (suponiendo que la red tiene un servidor DHCP):

```
auto eth1
iface eth1 inet dhcp
```

Asimismo, para contar con la conexión Ethernet integrada que será utilizada para la red interna de clúster, tienes que crear un archivo con el nombre eth0 en la misma carpeta y que incluirá una dirección IP estática:

```
auto eth0
iface eth0 inet static
  address 10.10.10.1
  netmask 255.255.255.0
  network 10.10.10.0
  broadcast 10.10.10.255
```

Es necesario configurar un servidor DHCP en el nodo maestro para poder proporcionar una dirección IP a los nodos esclavos de la red interna, y el nodo maestro tendrá que ofrecer servicios NAT entre las redes interna y externa. Por otra lado, todos los nodos necesi-

tarán tener editado su archivo `/etc/hosts` para permitir el direccionamiento nemo-técnico de los nodos por su nombre sin la necesidad de un servicio DNS. Las instrucciones detalladas para llevar a cabo estas tareas las pueden encontrar en mi blog en <http://bit.ly/2aJdAmi>.

Una vez que los nodos estén configurados para su conexión en red, se pueden apagar y desconectar de la red doméstica. La conexión Ethernet integrada de los nodos se debe conectar al switch Ethernet de la red interna y la red doméstica conectarse al adaptador Ethernet USB 3 del nodo maestro.

Antes de reiniciar cada nodo, formatea las tarjetas microSD con un sistema de archivos ext4 e insértalas en cada nodo. Inicia todos los dispositivos. Deberías ser capaz de conectarte via SSH al nodo principal, y desde allí conectarte a cada esclavo también via SSH. Tu tarea final es configurar el archivo `/etc/fstab` en cada dispositivo para que la tarjeta microSD sea montada en el punto de montaje `/data`. Para ello, es necesario encontrar el UUID del volumen de la tarjeta microSD después de montarla por primera vez con el comando `blkid`. Después, añade una línea en el fichero `/etc/fstab` como esta:

```
UUID=c1f7210a-293a-423e-9bde-
1eba3bcc9c34 /data ext4 defaults
0 0
```

Sustituye el UUID de tu tarjeta microSD por el que aparece más arriba. Una vez completados estos pasos, dispondrás de un clúster totalmente configurado que está listo para instalar un software big data como Hadoop. La instalación de Hadoop es un proceso bastante complicado, que cubriré en mi próximo artículo. Por ahora, hemos montado y configurado correctamente un clúster XU4 que podemos utilizar para cualquier tipo de procesamiento de datos complejos. Dispones de más información sobre este clúster ODROID-XU4 en <http://bit.ly/2aJdAmi>.

# XPOSED FRAMEWORK

## DOMINA LAS ACTUALIZACIONES DE ANDROID Y GESTIONA LOS CAMBIOS DE NIVEL DEL SISTEMA

por Jörg Wolff



**T**ratar la cuestión de hacer cambios de nivel en el sistema para tu sistema operativo Android. Se puede lograr a través de un laborioso proceso de creación de ROMs personalizadas, su instalación, trabajar en las actualizaciones y volver a instalarlas. ¡Algo insostenible!

Los entornos de trabajo como Xposed pueden salvarte en muchos casos, siempre y cuando el código/comportamiento original no se haya modificado demasiado. Teniendo acceso root, puedes hacer cambios en el nivel del sistema, sin necesidad de instalar ROMs personalizadas. Los cambios realizados se pueden deshacer simplemente desactivando el módulo y reiniciando.

### Instalación

Se ha utilizado un ODROID-C2 en este ejemplo. Para empezar, abre una ventana de terminal y crea un nuevo directorio de trabajo en el dispositivo, descarga la versión ARM/Lollipop del entorno de trabajo (`xposed-v84-sdk22-arm.zip`) desde <http://bit.ly/1VOp1wW>. Descomprime el archivo. Desde el mismo directorio de trabajo, descarga el apk instalador (`XposedInstaller_3.0_alpha4.apk`) desde <http://bit.ly/1GODrBg>.

### Ejecuta los siguientes comandos:

```
$ adb install XposedInstaller_3.0_alpha4.apk
$ adb shell
$ mount -o remount, rw /
$ mkdir tmp
$ cp -av sdcard/xposed-v84-sdk22-arm /tmp/
$ cd tmp/xposed-v84-sdk22-arm
$ cp -av META-INF/com/google/android/* .
$ chmod 755 flash-script.sh
$ ./flash-script.sh

...
*****
Xposed framework installer zip
```

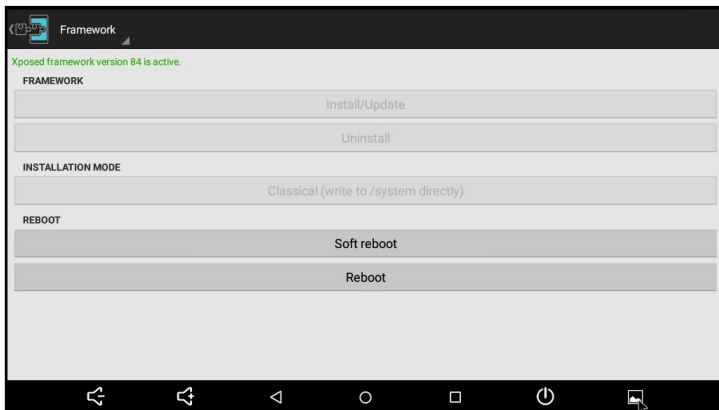


```

*****
- Mounting /system and /vendor read-write
- Checking environment
  Xposed version: 84
- Placing files
- Done

```

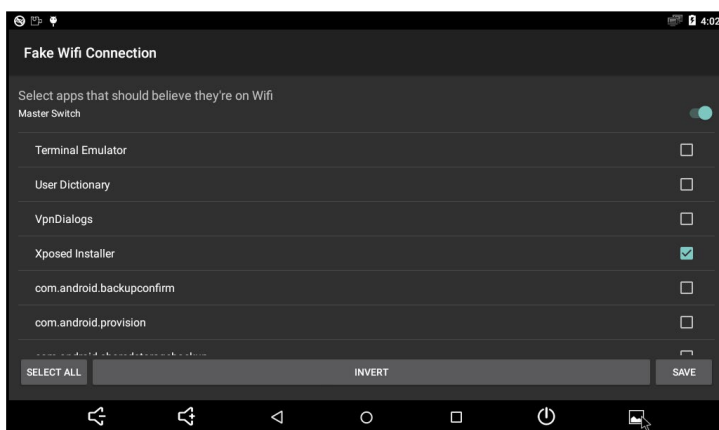
La pantalla de configuración de Xposed se muestra en la siguiente figura. Ten en cuenta que la opción de instalación/actualización está desactivada. En mi caso, puesto que me en-



**Xposed Framework UI**

contraba en la LAN cableada, sospechaba que la ausencia de una conexión Wi-Fi fue lo que causó que se desactivara esta opción. La herramienta Fake Wi-Fi Connection fue la que me ayudó a solucionar esa cuestión.

Descarga el apk Fake Wi-Fi desde <http://bit.ly/1rdfT7B>, a continuación, haz sideload usando la herramienta ADB. Inicia



**Configuración de Fake Wi-Fi**

Fake Wi-Fi Connection y selecciona el instalador Xposed. Tras completarse la instalación, debería ser capaz de ejecutar la opción Instalar/Actualizar en Xposed.

## Referencias

<http://bit.ly/2ao423N>      <http://bit.ly/1hCaq32>  
<http://bit.ly/1VOp1wW>      <http://bit.ly/1GODrBg>  
<http://bit.ly/1rdfT7B>

# BROTHERS: A TALE OF TWO SONS

## UN FANTASTICO JUEGO POR FIN EXPORTADO A ANDROID

por Bruno Doiche



La oportunidad de jugar a un juego que fue elogiado como una obra maestra, y que fue dirigido por un galardonado director no se ve todos los días, así que compruébalo por ti mismo y prepárate para disfrutar de un juego que antes sólo estaba disponible para consola..

Con algo de lo que creo que era una de las mejores historias de



**Te espera una gran aventura que desearás que nunca termine**

la última generación de juegos, me hace recordar el clásico de PS2, Shadow of Colossus. Te encuentras inmenso en una búsqueda que, aunque no llegas a entender el idioma de los hermanos, te sumerge en un mundo repleto de pequeños detalles e interacciones con el medio ambiente, los personajes y los objetos.

Si tiene hermanos, lograrás por ti mismo reconocer esa peculiar relación entre hermano mayor y menor. Este cariñoso contraste de responsabilidad y alegría me infunde un cierto sentido de admiración por la hermosa tierra nórdica en la que se inspira este juego.

<https://play.google.com/store/apps/details?id=com.and.games505.brothers>

# VOLUMIO 2.0

## TU FAVORITO REPRODUCTOR DE MUSICA INTEGRADO ESTA MEJOR QUE NUNCA

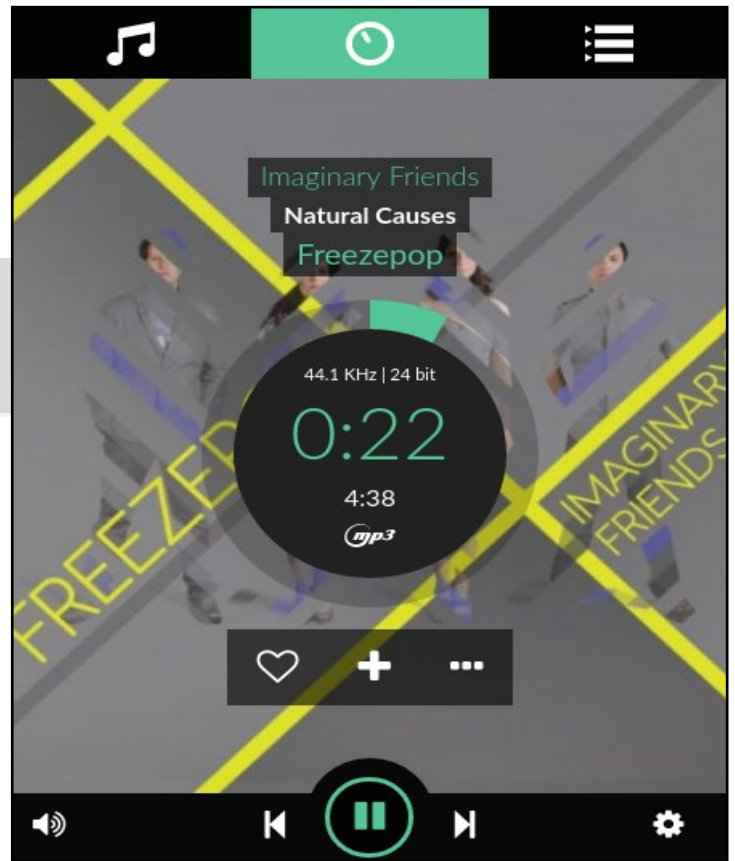
por @synportack24

Si no ha oído hablar de Volumio, se trata un de gran sistema operativo repleto de funciones con el que puedes convertir tu ODROID en un jukebox controlado por WiFi. Y si ya estás familiarizado con Volumio, puede que te estés preguntando por qué las cosas han estado muy tranquilas en lo que respecta a las actualizaciones.

Volumio 2 representa un avance muy importante y está lejos de ser una simple y nueva versión. Toda la interfaz web ha sido reescrita en Node.js, además le acompaña un sistema operativo totalmente adaptado. El resultado de estos cambios es una interfaz más modular que hace que desarrollar plugins sea mucho más sencillo. Otro de los principales cambios que notarás con respecto a la versión 1 es que es bastante más rápido. Casi todos los componentes de Volumio 2 parecen más rápidos. Se inicia en segundos y apenas hay retrasos cuando te mueves por la interfaz de usuario web. El sistema operativo está disponible para muchas plataformas ODROID, incluyendo el C0, C1, C1+, C2, XU3, XU4, y X2. Tienes el enlace a los binarios pre-compilados y al código fuente del proyecto al final del artículo.

### Configuración

Durante el último mes, he estado ejecutando Volumio 2 RC2 sobre un ODROID-C2 con un HiFi Shield, estoy esperando con mucha emoción la versión completa. La configuración de Volumio 2 es extremadamente simple y necesitas tan sólo unos cuantos clics para que las cosas empiecen a funcionar. Simplemente graba la imagen de Volumio en una tarjeta micro SD o módulo eMMC y conectar tu ODROID a la red vía Ethernet. Ten en cuenta que al principio tienes que utilizar la conexión Ethernet, en lugar del Wi-Fi, sólo para configurar las cosas. Abre un navegador y escribe "volumio.local". Volumio 2 utiliza un protocolo conocido como Bonjour, de modo que si tienes un navegador que lo soporte, no necesitas conocer la dirección IP del dispositivo. No he tenido problemas al usar esta dirección con iOS y en varios navegadores con Linux, pero en Android Marshmallow sólo funciona si introduces la IP del dispositivo. Al hacer clic en el icono de configuración "gear" de la esquina superior derecha aparece una lista de páginas de configuración. Para añadir una conexión Wi-Fi, haz clic en la pestaña "Network" y desplácese hacia abajo para localizar una lista de redes locales. Con la versión actual (RC2), me di cuenta que no todos mis adaptadores Wifi USB eran compatibles, aunque tuve suerte con el módulo 0 WiFi de Hardkernel.



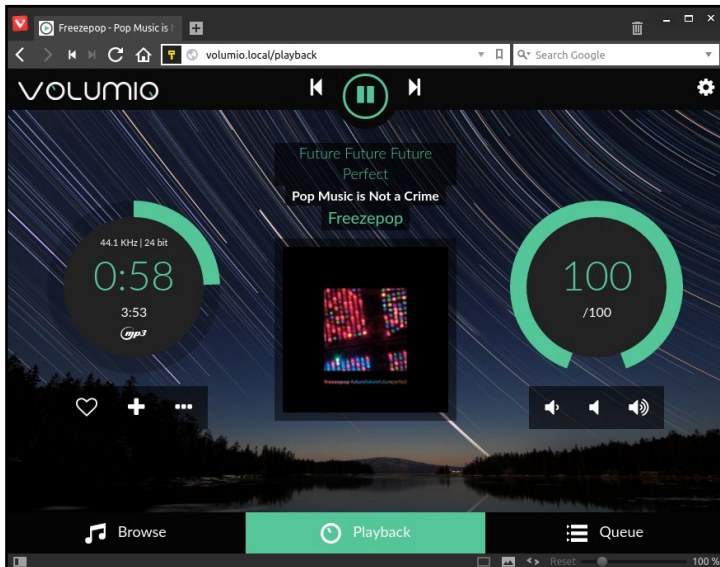
Volumio 2.0 fue reescrito en Node.js y es mucho más rápido

Tras finalizar la configuración Wi-Fi, puedes desconectar el cable Ethernet. El siguiente paso es añadir un NAS o alguna fuente de música. Configurar un NAS es extremadamente sencillo, ya que Volumio busca y enumera de forma automática todos los dispositivos de almacenamiento locales y en red. Una vez añadido un dispositivo, buscará y añadirá toda la música. Si cuentas con un HiFi shield, te recomiendo que cambies la salida de audio HDMI por Shield en la pestaña "Playback Options". ¡Eso es todo lo que necesitas para empezar!

### Análisis

En esta evaluación, ten en cuenta que yo he estado utilizando una versión candidata (beta), de modo que hay errores y problemas que habrán sido corregidos antes de la lanzamiento de la versión final. Incluso en el último mes ha habido un enorme progreso en muchas cosas. Las actualizaciones se pueden realizar fácilmente a través de OTA (por el aire) directamente desde la interfaz web.

No fui capaz de montar y escanear mi principal recurso compartido Samba, aunque un recurso compartido creado en mi portátil Debian me funcionó sin problemas. Estoy seguro



The new interface is easy to use and also includes an Android app

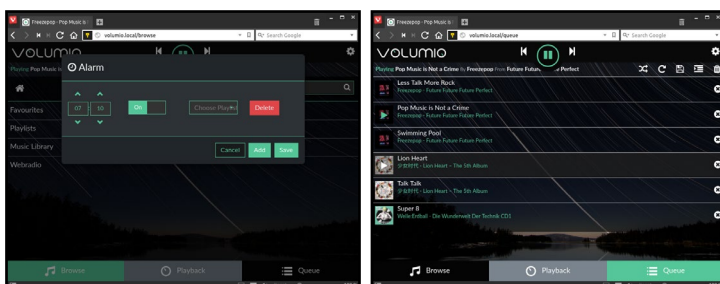
de que con un poco más de empeño, podría haber conseguido montarlo correctamente. Más allá de eso, Volumio fue capaz de reproducir todos los formatos que tengo. Los diferentes tipos de tasa de bits de archivos MP3, M4As, FLAC, OGG se reproducían estupendamente. Algunos archivos FLAC más grande necesitaría un segundo o más para el búfer, pero logré paliar este tema cambiando algunas opciones del búfer de archivos. Desde la página principal, puedes navegar fácilmente por la cola de música, buscar más canciones o simplemente ver lo que se está reproduciendo. Es importante que tengas tu música organizada de una forma lógica, ya que Volumio sólo utiliza tu estructura de directorios. La interfaz web se adapta muy bien tanto si utilizas una pantalla grande como un teléfono, y sigue siendo muy rápida incluso en los teléfonos más antiguos. También existe una app de Volumio para Android.

En conjunto, Volumio es un jukebox con multitud de funciones que es extremadamente fácil de instalar y configurar. Tras haber usado Volumio durante aproximadamente un mes, pude ver su enorme potencial. Estoy esperando con mucha emoción el lanzamiento final de la versión 2, es muy recomendable para cualquiera que busque un reproductor multimedia en red. Para obtener más información, o para descargar Volumio y probarlo por ti mismo, visita los siguientes enlaces:

**Volumio 2.0:** <http://bit.ly/2a5TxxN>

**Image download page:** <http://bit.ly/2aciw17>

**Github page:** <http://bit.ly/2aoJUfq>



# THAT LEVEL AGAIN

## DONDE TODOS LOS NIVELES SON IGUALES, PERO NO LA FORMA DE GANAR

por Bruno Doiche



En los juegos de puzzles normalmente tienes que descubrir un determinado patrón para resolverlos, y una vez que lo haces, ya no tendrán problemas para completar de nuevo el juego, ¿verdad?

Pues bien, That Level Again desmonta esta premisa, con 64 niveles y cada uno presenta 64 modos diferentes de resolverlo. Supone un desafío nunca antes visto en los juegos de puzzles.



Para aquellos que le gustan salirse de los convencionalismos será un gustazo, y para aquellos que sólo esperan un simple juego de puzzles, bienvenidos a un reto realmente difícil.



<https://play.google.com/store/apps/details?id=ru.iamtagir.game.android>

# CONOCIENDO UN ODROIDIAN

JOSHUA SHERMAN,  
EDITOR ADJUNTO DE ODROID MAGAZINE

editado por Rob Roy



Josh disfruta de la fotografía, así como de la informática



En Londres, probando las Google Glass con un colega

## *Háblanos un poco sobre ti*

Soy uno de los editores voluntarios de ODROID Magazine y me encanta experimentar con los dispositivos, siempre buscando nuevos proyectos para dar salida a tu dinero. Soy un exredactor de una publicación de tecnología llamada Digital Trends, y actualmente trabajo en el área de la bahía de San Francisco, aunque soy originario de Nueva York, en la costa este de los Estados Unidos. Aunque me gusta desmontar y trapichar con los gadgets, mi educación formal es la lengua Inglesa cursada en una universidad de Humanidades en Nueva York, lo cual contrasta con algunos de los candidatos a doctorado e impresionantes ingenieros que veo que contribuyen con esta impresionante revista. Esto no me impide romper y (en la mayoría de los casos) con el tiempo reparar equipos de todas las clases y tamaños en todo tipo de proyectos, incluidos los ODROIDS.

## *¿Cómo fueron tus inicios con los ordenadores?*

Mi padre me dio un ordenador cuando era muy joven, aunque “muy joven” para mí es un PC Pentium ejecutando Windows 98. En seguida encontré la forma de instalarle un arranque dual con Windows XP ya que estaba interesado en las nuevas características del XP, pero también quería mantener el soporte para juegos DOS de Windows 98. Desde entonces, me he vuelto más cauteloso, al convertirme en un seguidor incansable de la industria de la informática, trabajando como

periodista con teléfonos móviles e interesándome por cualquier cosa relacionada con las CPUs, ya sea sentado en mi mesa de trabajo, con un emulador arcade casero o en el bolsillo. Siempre he estado buscando formas de sacarle el máximo partido al hardware que tenía, y en el camino a menudo me unía a todo tipo de comunidades, como la comunidad de ODROID, XDA-Developers, PPCGeeks, entre otras, para compartir mis conocimientos y aprender de los demás. Esto es lo que llevó a escribir guías sobre cómo configurar una RTL-SDR en un dispositivo Android, convertir un ordenador en un emulador arcade y por supuesto contribuir a ODROID Magazine.

## *¿Qué te atrajo de la plataforma ODROID?*

Es muy versátil como plataforma de hardware, y atrae a una comunidad muy activa de creadores y desarrolladores. Empecé con la Raspberry Pi B+, pero pronto descubrí que su hardware me limitaba mucho a la hora de hacer lo que quería. Por aquel entonces, la emulación arcade y el ODROID-XU4 abrieron muchas más puertas para desarrollar mejores opciones de emulación y Bootstrap con más características que serían imposible de poner en práctica en el último hardware Pi, aunque depende en gran medida de la intensa labor de los desarrolladores del mundo de los ordenadores de placa reducida. En general, me gusta el hardware de gama alta como el XU4, la comunidad dedicada que tenemos, y el soporte directo que ofrece Hardker-

nel sobre nuevos productos y sistemas operativos.

*¿Cómo utilizas tus ODROIDS?*

Ahora mismo tengo un puñado de ODROIDs (un C2, un C1, y dos XU4s) que están en cajas ya que me acabo de mudar. Una vez que los saque, tengo pensado experimentar con el desarrollo de un dispositivo multimedia avanzado, probar un DVR en un ODROID y tal vez, incluso probar con la domótica. También he probado recientemente la emulación x86 realmente increíble con un XU4, aun sigo pensando en una solución de forma de servidor o firewall casero creado a partir de unos cuantos XU4s, ¡si consigo algunos más! También estoy pensando en un proyecto de estación meteorológica auto-sostenible con ayuda de un ODROID-C0, pero todavía tengo que conseguir muchas piezas para llevar a cabo esta tarea.

*¿Cuál es tu ODROID favorito y por qué?*

El ODROID-XU4 me impactó la primera vez que lo compré. Fue increíble ver la fluidez con la podía manejar Android, con un montón de capacidad de procesamiento para experimentar como dispositivo de sobremesa en toda regla. Siempre intento utilizar el más destacado y potente hardware que nos ofrece el mundo de los ordenadores de placa reducida.

*¿Qué innovaciones te gustaría ver en futuros productos de Hardkernel?*

Soy un gran fan del WiFi y del Bluetooth integrado para el próximo XU4, lo cual ayudaría a reducir el número de puertos USB dedicados a los periféricos, sobre todo cuando me gusta

empezar con un teclado y un ratón para algunos proyectos experimentales, o para mi potente servidor multimedia, aunque prefiero minimizar la necesidad de usar otros cables o hubs USB externos. Lo más importante, espero que Hardkernel encuentre la forma de mejorar el rendimiento de la GPU y la VPU, ya sea a través de los kernels, hardware o soporte para Tegra u otras potenciales necesidades de procesamiento gráfico basados en ARM. También espero que la placa meteorológica reciba una actualización para soportar sensores externos, o algún otro modo de separar los componentes electrónicos del entorno para lograr un sistema de vigilancia exterior más robusto.

*¿Qué aficiones e intereses tienes aparte de los ordenadores?*

Los pequeños y grades Ordenadores, incluyendo teléfonos inteligentes, ordenadores de sobremesa y SBC, absorben la mayor parte de mis aficiones, aunque me gusta la fotografía con mi Sony A6000, montar en bicicleta y escribir. También tengo un VPS Linux con el que me encanta probar nuevas aplicaciones web o simplemente experimentar con el último software en la nube privado.

*¿Qué consejo le daría a alguien que desea aprender más sobre programación?*

Todavía soy un completo novato en lo que respecta a la programación, pero sin duda recomiendo probar algunos experimentos o proyectos que aparecen en los foros ODROID y en otros lugares. Ofrecen muy diversas versiones del sistema operativo Linux, que permiten editar scripts shell y crear tus propias secuencias de comandos shell para cualquier cosa que necesites. Esto es sólo un comienzo, creo que el conocimiento de los demás es el mejor lugar para empezar.

**Josh y su madre en París, visitando la Torre Eiffel**



**Con el perro de la familia, un Pinscher miniatura llamado Trixie**

