Tuxinfo

/tuxinfo
@tuxinfo

una revista libre, para un mundo libre.

nro.57

Kali LINUX
el nacimiento

Cuando deja de ser un chiste
opinión

Tiny Tiny RSS
(bye bye Google Reader)

OpenOffice tutorial
parte II

Redes para las masas
Parte II

myplex
música en linea multiplataforma

Fedora 18
tutorial de instalación

Chromium OS
Un vistazo a la versión libre del OS de Google

M4
Mobile Malware for Mobile Money

Probamos una Ultrabook Bangho Zero con Linux

Cine Mudo en Blender

Almacenamiento Híbrido

Martín Escoyez | LibreGraphics/Crafts

#57
abril 2013

Esta revista se publica bajo una licencia de **Creative Commons CC BY-SA 3.0** Puedes copiar, distribuir, mostrar públicamente su contenido y hacer obras derivadas, siempre y cuando **a)** reconozcas los créditos de la obra y **b)** la compartas bajo la misma licencia.

Microsoft, Apple, Sun, Oracle, así como otras marcas comerciales mencionadas en esta revista son propiedad de sus respectivas empresas.

Dirección
Ariel M. Corgatelli

Marketing
Claudia A. Juri

Corrección
Luis Luque
Oscar Reckziegel

Diseño de tapa
Martín Eschoyez

Diseño
Tuxinfo

www
http://www.tuxinfo.com.ar

facebook
http://www.facebook.com/tuxinfo

email
info@tuxinfo.com.ar

twitter
@tuxinfo

editorial

Otro nuevo número de TuxInfo en la red, y como siempre con mucha información. Como primer tema, quiero contarles que estaré de forma personal haciéndome cargo de la maquetación de la revista, (con lo cual tengan paciencia con la estética). También tenemos una incorporación al staff, este mes se suma Jorge Mieres, quien se desempeña como consultor de seguridad informática en Argentina para una empresa importante como lo es Kaspersky. Jorge estará escribiendo mensualmente una columna de seguridad, la cual tendrá, por supuesto, relación con el mundo del software libre.

Volviendo a la revista, este mes hubo bastante movimiento, y quizás algunas de las noticias que hubo, serán las más importantes del año. El mes pasado hablábamos del próximo Nexus 5, lo cierto es que hasta ahora no hay definición de Google al respecto; lo que sí vemos, es un interés un poco más fuerte por utilizar la maquinaria de Motorola Mobility, algo que nos parece muy bien desde este lado. Con lo cual seguramente tendremos algún Nexus de Motorola muy pronto.
Luego otra noticia que ha dado qué hablar, fue la que protagonizó Richard Stallman, realizando un pedido a las autoridades del FLISOL 2013, donde los instaba a que no recomienden el uso de Ubuntu dentro del evento. La noticia no cayó del todo bien y hubo por supuesto muchas críticas. Finalizado el tema, RMS tuvo que explicar sus dichos, aclarando que él no obligaba a los organizadores a prohibir Ubuntu, sino que les pedía que no hagan difusión de la misma.

Pero como siempre, en nuestra revista podrán encontrar mucha información y temas varios para deleitarse con los mejores artículos de la única revista que apuesta, sin importar el rédito económico, al software libre, Linux y Android.

Como para redondear el editorial, les cuento que el número está plagado de notas interesantes tales como: Kali LINUX - el nacimiento; Tiny Tiny RSS (bye bye Google Reader); OpenOffice tutorial - parte 2; Redes para las masas - Parte II; myplex música en línea multiplataforma; Chromium OS - Un vistazo a la versión libre del OS de Google; Probamos una Ultrabook Bangho Zero con Linux; Cine Mudo en Blender; y mucho más...

Y como todos los meses, repetimos la misma convocatoria en donde podamos tener más sugerencias de ustedes y así adaptar los contenidos de las notas a vuestras necesidades y preferencias, las mismas las podrán realizar a nuestros medios de contactos.

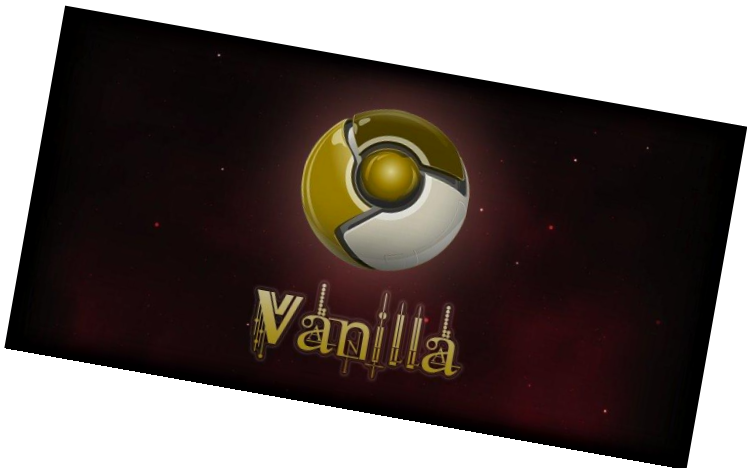
Fan page:
https://www.facebook.com/tuxinfo
User Twitter: @tuxinfo
Mail de contacto:
info@tuxinfo.com.ar

¡Sigan pasando la voz!
Hay otro nuevo número de TuxInfo para descargar de forma gratuita.

Ariel M. Corgatelli
@arielmcorg

índice

04- Chromium OS Un vistazo a la versión libre del OS de Google.
07- Guía de OpenOffice - parte 2.
09- Cine Mudo en Blender.
12- Redes para las masas - Parte II.
24- Cuando deja de ser un chiste.
25- Almacenamiento híbrido.
27- Kali LINUX El nacimiento.
29- Probamos una Ultrabook Bangho Zero.
31- Virtualizando con Fedora una guía básica sobre cómo crear VM's.
37- Tiny Tiny RSS - (bye bye Google Reader).
41- M4. Mobile Malware for Mobile Money.
43- Myplex música en línea multiplataforma.



Chromium OS

Un vistazo a la versión libre del OS de Google

POR **Juan Manuel Dansa**

En esta oportunidad, le echaremos una mirada a la versión de código abierto del sistema operativo de Google: Chromium OS. Como ya se sabe el código fuente del mismo fue liberado en noviembre de 2009 bajo una Licencia BSD; fue construido sobre la base de un núcleo GNU/Linux, en un entorno Ubuntu 10.04, utilizando el gestor de paquetes oficial de la distribución GNU/Linux Gentoo, Portage. Este suceso es el punto de partida para la descarga y desarrollo del código fuente y el nacimiento de Chromium OS. La versión más conocida, la cual usaremos, fue creada por Liam McLoughlin, publicada bajo el nombre de Hexxeh, éstas compilaciones arrancan desde una memoria USB, Vmware o VirtualBOX e incluyen características como el soporte para el lenguaje de programación Java; su nombre clave es "Vanilla" siendo una de las únicas, sino la única, versión que podemos probar y utilizar. Hay que tener en cuenta que esta versión, con diferencia de la de Google, no posee soporte para Flash/MP3/PDF, ya que requieren de licencias para ser distribuidos, en el caso de necesitar la instalación de éstos, en el siguiente link encontrarán los scrips https://gist.github.com/dz0ny/3065781; otras cosas a tener en cuenta, es que es una versión NO oficial de Chrome OS y no representa la totalidad de la experiencia de éste, cada persona lo utiliza bajo su responsabilidad, no se actualiza de forma automática (hay que ir descargando de forma manual) y no todos los equipos poseen el soporte de hardware.

Lo primero que tenemos que hacer es dirigirnos hacia la página del desarrollador (http://chromeos.hexxeh.net/) y descargar la imagen en el soporte que deseemos, por ejemplo en versión de llave USB, (la cual creo que lo transforma en una interesante herramienta para su utilización o prueba en diferentes equipos), versión VMWare y VirtualBOX.

Instalación

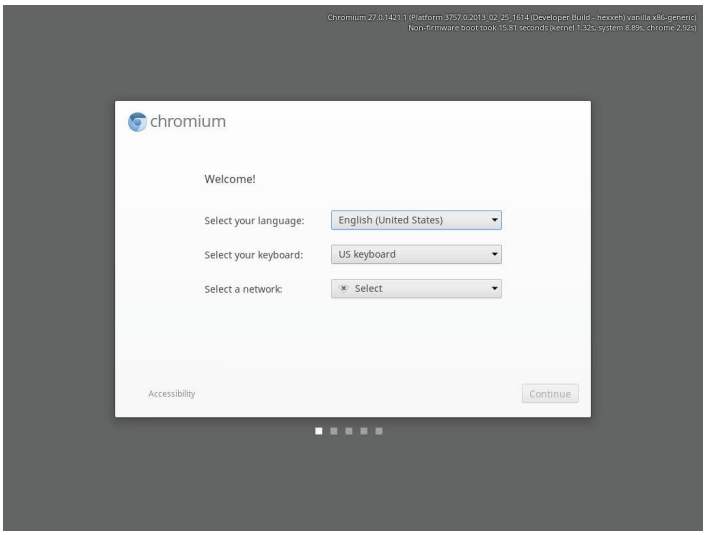
Para la memoria USB (como mínimo de 4GB de espacio), descargar el archivo ".zip", descomprimir y en:

LINUX:
Mismo directorio de la IMG:
dd if=ChromeOS.img of=/dev/sdX bs=4M

WINDOWS:
Pueden descargar el Windows Image Writer (http://sourceforge.net/projects/win32diskimager/files/latest/download) y escribir la IMG en el USB.

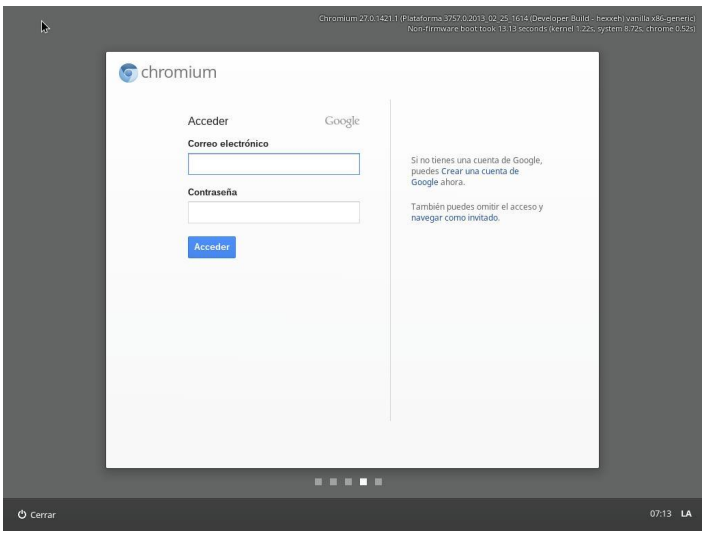
VirtualBOX:
En el caso de que utilicen una imagen de VirtualBOX, configúrenlo como una distribución GNU/Linux, les recomiendo Ubuntu, y en configuraciones de red, fíjense que se encuentre en NAT y en avanzadas, tipo de adaptador, pongan "red paravirtualizada (virtio-net)"; de lo contrario no reconocerá la conexión de internet. Otra recomendación a tener en cuenta es que si la integración del mouse falla, deshabiliten ésta desde el menú.

Dentro del proceso de instalación nos encontramos con unos pasos muy simples para su instalación.



Instalación - Elección de Lenguaje / Teclado / Conexión

Lo primero es la elección del idioma, el tipo de teclado y la red, siendo esta última la más importante, ya que no se podrá continuar con la instalación sin conexión a internet, por la simple razón que es un sistema ON-LINE y debe utilizar la cuenta de mail de Google que cada uno tenga para su sincronización.



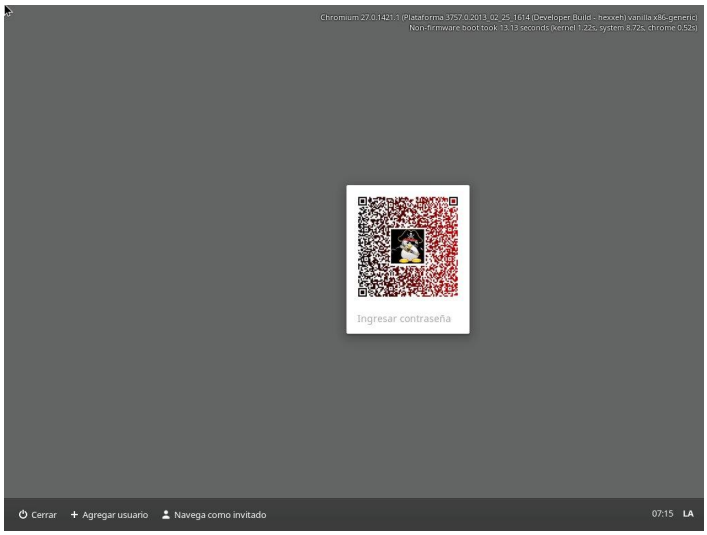
Primer pantalla - Presentación / Guía de uso

Ingreso a cuenta de Google

La carga es rápida y automática llevándonos, tras su finalización, a la pantalla de ingreso de cuenta y luego a la presentación/guía de uso. Cabe destacar la velocidad de ingreso, esto se debe a que según los documentos de diseño preliminar, Google describe una arquitectura de tres capas: firmware, navegador web, gestor de ventanas, y un nivel del sistema de software de servicios en espacio de usuario; siendo el firmware el que contribuye con el tiempo de arranque rápido ya que no se estaría comprobando el hardware; a su vez encontramos la presencia de la seguridad mediante la verificación de cada paso en el proceso de arranque y la incorporación de la recuperación del sistema.

Interfaz de Usuario

El sistema utiliza como interfaz de trabajo el propio navegador Chromium/Chrome y el conocido sistema de pestañas (tab) para levantar cada aplicación.

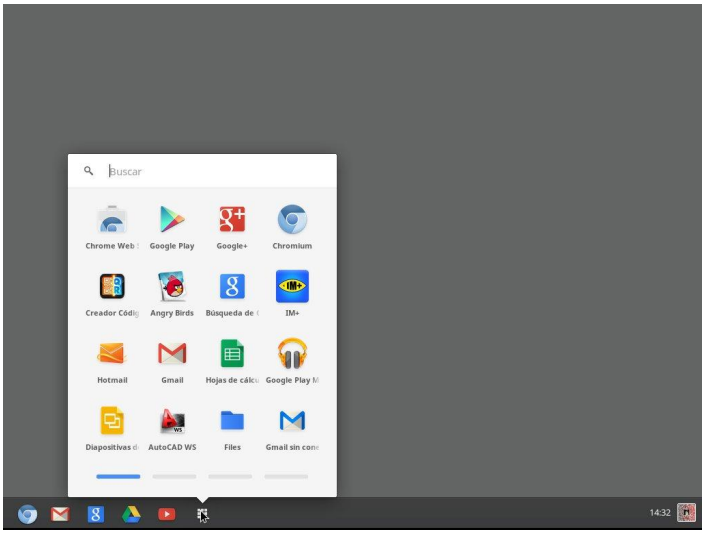


Pantalla de logueo minimalista.

Todo esto contribuye a un minimalismo muy prolijamente tratado, desde la pantalla de ingreso al sistema hasta el entorno de trabajo.
Lo más interesante es que si ya estamos utilizando el navegador Chrome/Chromium en cualquiera de las plataformas para las que se encuentra disponible, nos traerá todo lo que tengamos configurado en éstos, replicando en todos a la vez, respetando desde los programas hasta extensiones.

La forma en la que se presenta el menú es al estilo barra inferior con accesos directos, siendo el primero, obviamente, el navegador y

al final a la derecha la hora e ícono de usuario, donde al clickear se nos abre, entre otra información, el menú de configuración, cierre de sesión, sonido, etc.
Las aplicaciones son descargadas, como en los navegadores, desde el Chrome Web Store, dándonos un amplio popurrí de ellas, que en último tiempo han crecido en número rápidamente.



Barra del menú / Aplicaciones / Entorno

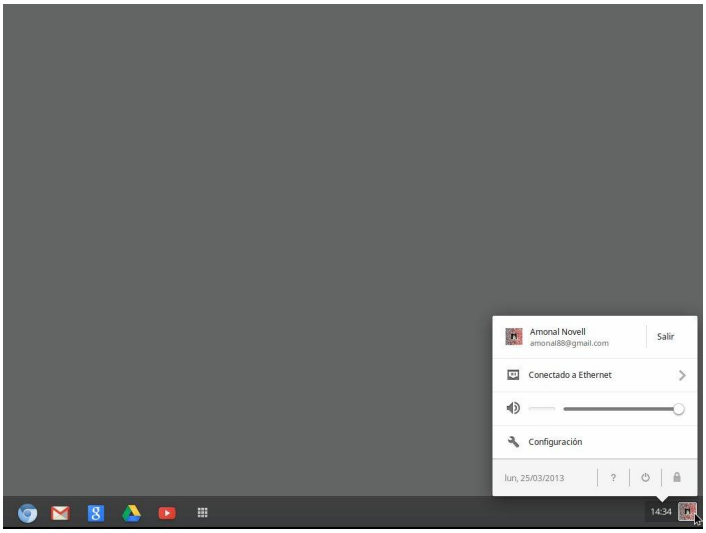
Otro dato interesante, es que posee un administrador de archivos el cual nos muestra lo que tenemos en nuestro espacio físico y lo que se encuentra en Gdrive, estando éste completamente integrado con el sistema.
Otro punto a favor, es que la interfaz es muy intuitiva y fácil de navegar, es prácticamente como utilizar el navegador con algunos pequeños ajustes para su utilización como sistema operativo.

Proyecto

Para los que deseen interiorizarse más con el proyecto, compilarlo e incluso desarrollar:
http://www.chromium.org/chromium-os,
encontrarán también todo tipo de guías y herramientas necesarias.

Conclusión

En los últimos tiempos vemos cómo se van dirigiendo los sistemas operativos hacia la completa integración con internet, “la nube”, y éste es la caso más representativo de todos, ya que prácticamente, depende por completo de una conexión para subsistir. Se nota una excelente apuesta de “la gran G” en este sistema, el cual se encuentra acompañado de equipos de excelentes prestaciones como las ChromeBooks, que poseen un bajísimo consumo logrando autonomías no alcanzadas todavía por equipos convencionales; a ésto le podemos sumar su velocidad al iniciar y su facilidad de uso como así también su bajo costo, llevando estas características a posicionarlo en un ambiente por ejemplo educativo, (donde más consumo va teniendo por el momento el producto) y con sólo pensar en estudiar una jornada completa sin recargar el equipo, evitándonos el transporte de baterías, es una propuesta muy difícil de rechazar.
Es claro que no es un reemplazo de Android y que una fusión es prácticamente imposible, pero es una excelente alternativa que nos da la empresa y en este caso los desarrolladores del proyecto Chromium OS, que a decir verdad me hace pensar cómo se vería mi Asus Transformer TF300T con este sistema y si, tal vez, sería más productiva.



Menú de usuario / Configuración / Cierre de Sesión

Juan Manuel Dansa (Amonal)
amonal88@gmail.com
twitter: @Amonal_
g+: Amonal Novell

Guía de OpenOffice.org

POR **RAFAEL MURILLO**

Guía de OpenOffice

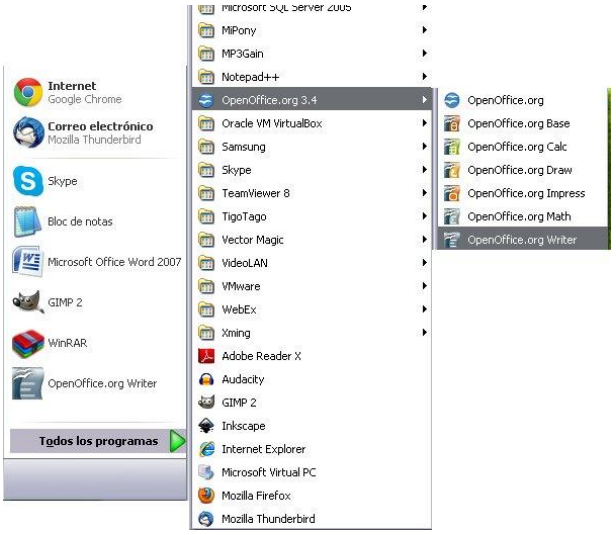
En el número anterior de esta guía, revisamos un poco sobre qué es OpenOffice.org, cómo y dónde descargarlo y cómo instalarlo. Ahora toca el turno de conocer poco a poco las distintas “herramientas” que lo conforman, para ello, vamos a iniciar con **Writer**.

OpenOffice Writer es parte de la suite de ofimática **OpenOffice**. Esta herramienta **es un potente procesador de texto**, **totalmente compatible con Microsoft Word** y además nos provee de nuevas funcionalidades que iremos descubriendo a lo largo de esta guía. Como ya sabemos, toda la suite Ofimática (me refiero a OpenOffice.org) **es multiplataforma**, esto quiere decir que puede correr en cualquier sistema operativo actual (GNU/Linux, Mac OS X, FreeBSD, Microsoft Windows, etc.) y **cuenta con traducciones para una gama muy variada de idiomas**.

Bien, una vez instalado, dentro del menú de Inicio (en el caso de Windows), en Todos los programas, veremos que se ha creado una nueva entrada para OpenOffice.org, es ahí donde se encuentran todas las herramientas que lo conforman. Vamos a abrir OpenOffice Writer dando clic en él, dentro del menú antes mencionado (para Linux, se creará una nueva entrada en el menú de Aplicaciones que se llamará Oficina o bien, Ofimática).

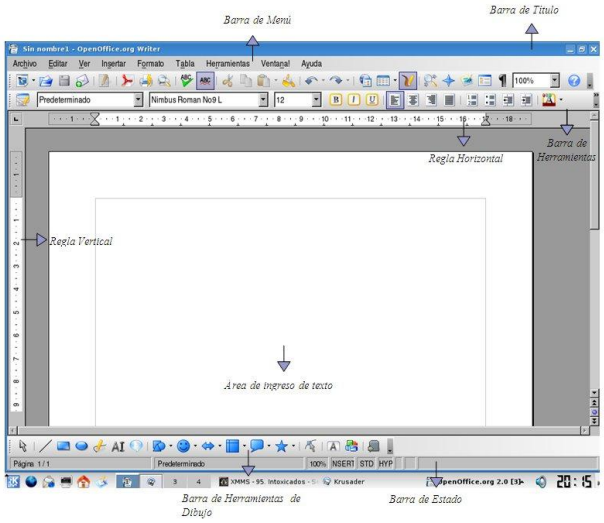
Una vez abierta la aplicación, nos vamos a encontrar con la siguiente pantalla (entorno de trabajo de OpenOffice Writer):
Como podemos ver en la imagen anterior, Writer cuenta con los siguientes elementos:

- Barra de título
- Barra de Menú
- Barra de Herramientas
- Barra de Estado
- Regla Horizontal
- Regla Vertical
- Área de ingreso de datos



Ahora que conocemos los elementos que conforman a la herramienta Writer de OpenOffice.org, vamos a describir con un poco más de detalle lo que es y para qué nos sirven.

La **Barra de Título contiene el nombre del archivo actual de trabajo y el nombre de la aplicación** que lo trata y en este caso es **Writer (**La barra de título estará presente en cualquier programa, no es exclusiva de OpenOffice.org**)**. También encontramos los controles básicos de manejo de la aplicación (**minimizar, maximizar y cerrar**).



En la **Barra de Menú encontraremos todas las opciones del Writer.** A continuación se detallan los menús que hay y un poco acerca de lo que encontraremos en ellos (posteriormente podremos ver con detalle cada opción, es recomendable que el usuario vaya entrando en las opciones para que se familiarice con ellas):

- **Archivo,** aquí podemos crear, abrir, guardar, imprimir, cerrar y exportar documentos.
- **Editar,** que tiene las opciones de copiar, cortar y pegar texto; también podremos buscar y reemplazar texto dentro del documento activo de trabajo.
- **Ver,** esta opción nos permite mostrar las barras de herramientas, la barra de estado, las reglas, los caracteres no imprimibles y como punto importante podremos definir la escala de visualización del documento (zoom).
- **Insertar,** nos permitirá ingresar saltos manuales, símbolos, campos, links, notas al pie, encabezados y tablas.
- **Formato,** desde la cual podremos aplicar formato al texto, como por ejemplo, aplicar estilos, espacios, sangrías, numeración y viñetas, convertir de mayúsculas a minúsculas y viceversa, columnas y más opciones.
- **Tabla,** con la que podremos crear, modificar y personalizar tablas.
- **Herramientas,** en donde tenemos opciones para comprobar ortografía y gramática, contar palabras, cambiar el idioma y demás.
- **Ventana**, este menú nos permite ir cambiando entre los diversos documentos que tengamos abiertos al mismo tiempo.

Además de la Barra de Menú, contamos con una **Barra de Herramientas** desde la cual podemos utilizar las funciones más importantes de la aplicación **a manera de atajo** en vez de ir directamente a los menús que explicamos con anterioridad, con el objetivo de ahorrar tiempo (como pueden notar, existe una marcada similitud entre Writer y Word, precisamente para hacerle al usuario más sencillo su paso de la suite de Microsoft a OpenOffice.org).

Al igual que en Word, en Writer contamos con una **Regla Horizontal** y una **Regla Vertical**, las cuales dan una ubicación en pantalla para alinear gráficos y utilizar tabulaciones además de ajustar los márgenes de página.

La Barra de Estado nos brinda información sobre el formato de texto actual, la página donde nos encontramos y si estamos en modo de Escritura o de Reemplazo.

El Área de ingreso de texto es el lugar principal donde escribimos nuestro texto e insertamos imágenes o damos formato al mismo (es la hoja en blanco).

Finalmente en la **Barra de Herramientas de Dibujo** contamos con opciones para insertar distintos gráficos, cuadros de texto y demás.... Como podemos darnos cuenta, es una interfaz muy parecida a la de Word de Microsoft, pero con muchas otras bondades que Microsoft no ofrece o no implementa de manera correcta.

Si bien esto es un vistazo a lo que es la herramienta de edición de textos de OpenOffice.org, el usuario podrá familiarizarse rápidamente con ella ya que la interfaz es muy intuitiva, además de que claramente está pensada para que, cualquier persona que lleve mucho tiempo de utilizar Word, de Microsoft, pueda cambiarse a OpenOffice.org sin ningún problema... Encontrará que los menús y opciones son muy similares, lo mismo que los íconos para llevar a cabo las tareas más comunes en la edición de sus textos.

En el siguiente número de esta guía, tocará el turno de echar un vistazo a **Calc**, el editor de Hojas de Cálculo de OpenOffice.org, ya verán que es muy sencillo de usar, y posteriormente nos adentraremos con más detalle en cada uno de los elementos que conforman la suite de OpenOffice.org.

Rafael Murillo
@linxack
linxack@gmail.com
www.itxperts.mx

Cine Mudo en Blender

POR **Ema Timbis López**

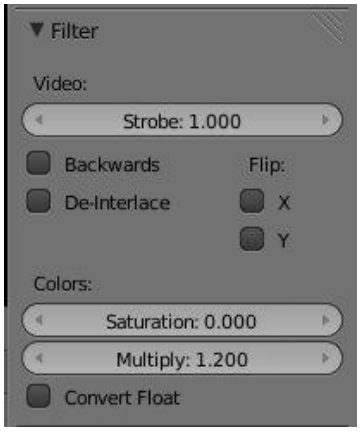
Si bien no creo que sea muy útil aprender a crear una película estilo mudo en Blender, me pareció buena idea para **aprender algunas funciones del Sequencer** (editor de video incorporado), e introducirlos en lo que es el pensamiento lateral, que ayudará en el uso de las herramientas más simples para crear efectos complejos.

Aconsejo ver algunas películas de Chaplin y/o del 1900 para saber de qué se habla y encontrar inspiración.

Lo primero que vamos a hacer es **añadir el video** con el cual vamos a trabajar. Ya en la ventana del *Video Sequencer Editor* (ver imagen 1), ponemos *Add>Movie*.

Blanco y negro

Vamos a desaturar el video, esto va a hacer que nuestra película se vea en blanco y negro, y multiplicar la imagen para que las partes blancas brillen aun más (da esa sensación de viejo). **Ajustamos *Saturation* a 0 y *Multiply* a 1.2**, ambas están en la sección *Filter* de la barra derecha (ver imagen 2).



Cambio de escena

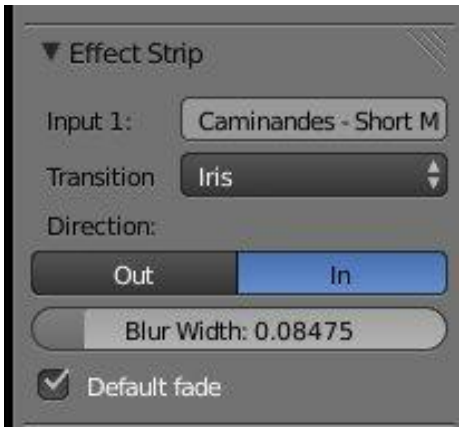
Los cambios de escena usuales eran donde la imagen se descubría en forma de círculo, o se escondía cuando se cerraba (no sé cómo lo hacían, pero posiblemente tenían un obturador). El efecto *Wipe* es perfecto para eso.

1- Necesitamos que un plano negro desaparezca y aparezca sobre el video, así que buscamos y **añadimos el plano** en *Add>Effect Strip>Color.*

2- Seleccionamos el video y luego la placa negra y **aplicamos el efecto** en *Add>Effect Strip>Wipe.* Se va a añadir un strip marrón, ese es el efecto.

3- Seleccionamos sólo el efecto *Wipe* y nos dirigimos a la sección *Effect Strip* de la barra derecha.

4- Cambiamos el tipo de **transición a *Iris*, la dirección a *In* y subimos el valor Blur Width** para que difumine el borde. (ver imagen 3)

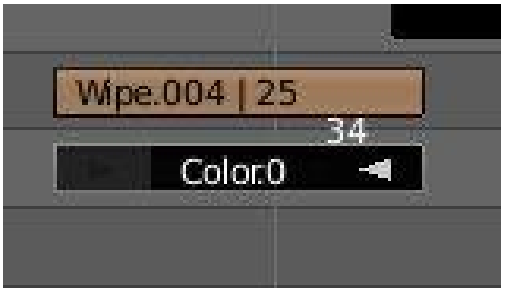


Efecto vignette

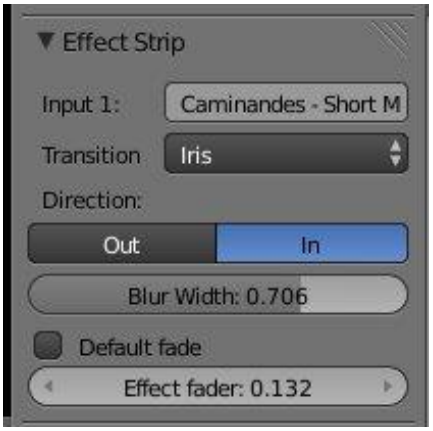
Cuando la luz pasaba por la diapositiva, solía focalizarse en el centro, lo cual dejaba un degradado hacia los costados. Nosotros vamos a volver a usar otra vez el efecto *Wipe* para eso.

1- **Seguir los tres pasos** de Cambio de escena.

2- Agarramos el plano negro de los lados y lo alargamos hasta que **cubra todo el largo del video**. Para ello presionamos botón derecho sobre uno de los costados del strip color (ver imagen 4), presionamos G y arrastramos.



3- Como no queremos que sea una transición, le quitamos la animación **desmarcando *Default Fade*** (en la sección *Effect Strip)* y **subimos el *Effect Fader*** el cual determina la apertura (ver imagen 5).



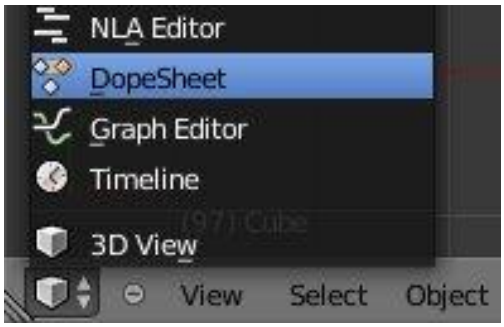
Saltos de la cinta

La cinta de la película cuando pasaba por el carretel solía "saltar", por eso las placas de texto o los títulos parecen animados cuando en realidad son sólo una imagen. Lo que vamos a hacer es que el video suba por la pantalla unos segundos y luego baje.

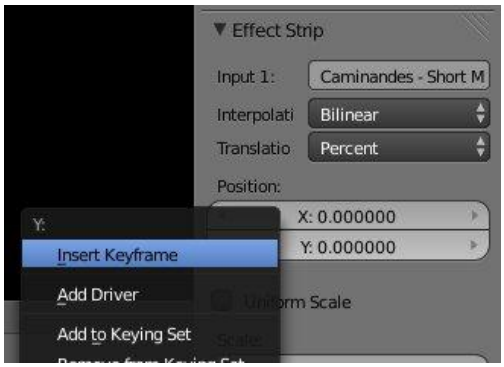
1- Hay que **transformar el video,** así que lo seleccionamos y ponemos *Add>Effect Strip>Transform.*

2- Seleccionamos el efecto (en verde). En la sección *Effect Strip* tenemos la opción de posición, vamos a **trabajar con el Y** que es el eje vertical.

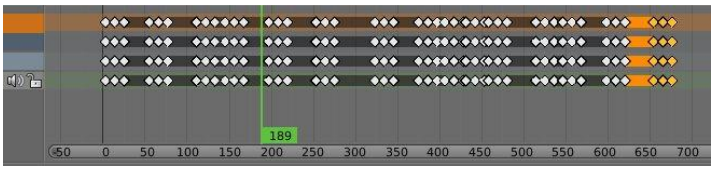
3- Abrimos una ventana de **DopeSheet** (ver imagen 6) y nos posicionamos en el **frame donde comienza el video**.



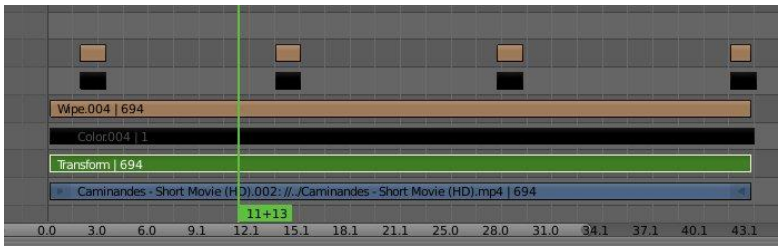
Presionamos botón derecho sobre el selector de Y **clickeamos en *Insert Keyframe,*** se volverá amarillo y aparecerán unos rombos naranjas en la ventana DopeSheet (ver Imagen 7).



4- Nos adelantamos unos 10 frames (haciendo click en la línea del tiempo del DopeSheet), ponemos **el selector Y en 6 e insertamos keyframe** nuevamente.
5- Adelantamos otros 10 frames, ponemos **el selector Y en 0 e insertamos keyframe**.
6- Haciendo click derecho en las tres columnas presionando la tecla shift, vamos a seleccionar los keyframes añadidos, luego con Ctrl + D los **duplicamos y los distribuimos por la película** (nos ahorramos el trabajo de añadirlos a mano cada uno). Obtendríamos algo similar a la imagen 8.



Placas de texto, marco y música.
La manera de añadir otros componentes a la película es bastante simple, pues ya **conocen el menú *Add***. Sólo recuerden que todas las imágenes que añadan estén **por encima de los efectos**, y que apliquen sus propios efectos *Wipe* y *transform.*



Espero que les haya gustado y si no encuentran algo, o no se entiende la explicación, pueden comunicarse conmigo a través de Twitter.

Ema Timbis López
@timbislopez

#Radiogeek
Podcast Diario
de Tecnología
www.radiogeek.ivoox.com

Redes para las masas - Parte II

POR **Hernán "HeCSa" Saltiel**

Introducción

En el artículo pasado comenzamos a estudiar algunos conceptos generales de comunicaciones. Entendimos qué es un emisor, un receptor, un mensaje, sus medios de difusión, y de una forma lo más humana y amena posible nos introdujimos en una de las formas de diferenciar las redes de datos según su topología física.

En esta nueva entrega, veremos cómo se diferencian según su alcance, y expondremos ejemplos que nos harán la vida un poco más sencilla sobre los distintos elementos de red que veremos en nuestra caminata por el mundo de las redes de datos.

Clasificación de redes de datos según su alcance

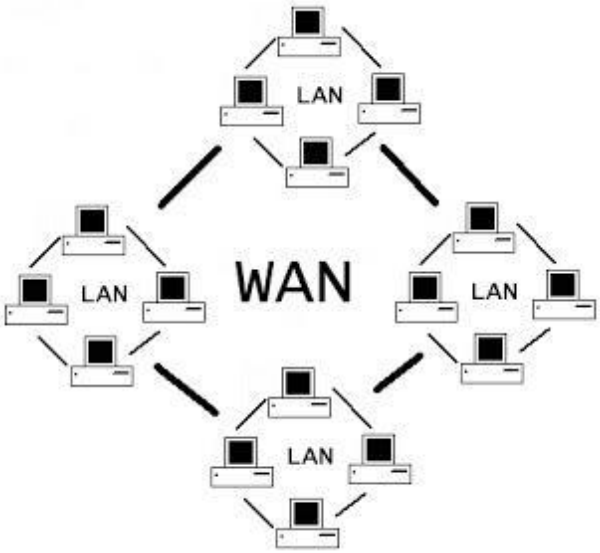
Cuando analicemos redes de datos según su alcance, tengamos en cuenta que salvo casos donde las diferencias son en extremo claras, hay varios en los que la frontera que existe entre las de un tipo y las del que le sigue en cuestión de distancias puede estar algo confusa. Es en estos casos en los que la tecnología utilizada para la interconexión de los sistemas de dicha red puede marcar la verdadera diferencia.

Habiendo salvado este punto, veamos cómo las redes se diferencian según su alcance.

- Red de área local: Las redes de área local son también conocidas, dadas sus siglas en inglés, como "LAN" (Local Area Network). Originalmente se consideraba locales a las redes que interconectaban

sistemas dentro de una misma sala, y en el mejor de los casos, en el mismo edificio. Pero el tiempo, y nuevos dispositivos de interconexión tales como bridges, repeaters, y otros tantos cambiaron este concepto para que primero se piense en extensiones de un kilómetro, y luego de más (las redes de fibra pueden llegar a los 200 Km, por ejemplo) en lo que a espacio físico entre sistemas conectados respecta. Éste es uno de los casos donde el tipo de interconexión y el direccionamiento IP puede marcar la verdadera diferencia. Más adelante veremos de qué se tratan estos conceptos.

- Red de área ámplia: Este tipo de red, dadas sus siglas en inglés, es conocida como “WAN” (Wide Area Network). Sobre la distancia que puede cubrir aún hay acaloradas discusiones, ya que se pretende pensar en unos mil kilómetros de separación entre sistemas, pero como antes mencionamos, será el direccionamiento IP y la tecnología de interconexión lo que determine si la red es o no WAN. Si miramos la parte de atrás de nuestros modems de conexión a internet, notaremos que una de las bocas de red, específicamente la que nos conecta a Internet, tiene esta sigla. Eso se debe a que interconectan la red local de nuestra casa, o empresa, o sencillamente nuestra computadora, con Internet. En Internet el rango de direcciones IP será diferente al de nuestra LAN, así como de seguro los medios de interconexión utilizados. Es muy común que se haga uso de sistemas de interconexión públicos para este tipo de redes (red de telefonía, de televisión por cable, etc.).



- Red de área personal: Como en los casos anteriores, su nombre derivado de sus siglas en inglés es “PAN” (Personal Area Network) y no tiene nada que ver con lo que usamos para hacer nuestros sandwiches. La idea de este tipo de redes es que interconecten dispositivos que están al alcance de una misma persona. Para clarificar el caso con un ejemplo, si tenemos un teléfono celular al que le deseamos conectar un auricular Bluetooth, la red establecida entre dichos dispositivos es

considerada una PAN. Si queremos pensar en distancias y coberturas, lo que cubra nuestra voz (si te llamás Pavarotti o Marcel Marceau no estarás representado en este ejemplo) será la distancia que una red PAN podrá cubrir. En general, son inalámbricas.

- Red de área local inalámbrica: De sus siglas en inglés recibe el nombre de “WLAN” (Wireless Local Area Network). Cuando las personas que implementaban redes de cobertura local se cansaron de cablear como locos edificios enteros, taladrar paredes para pasar cables, y colocar conectores en los lugares más inesperados (he llegado a ver conectores de red en baños, al lado de rollos de papel higiénico), descubrieron que si un locutor podía hacer pasar su ronca voz usando una onda de radio también podrían, ellos, pasar bits y bytes. Así nació la red inalámbrica de cobertura local.

Lógicamente, la frecuencia que se usa en el caso de las redes de datos es mucho mayor. También lo son los peligros a los que una red de este estilo está expuesta, ya que el tener ni más ni menos que el aire como medio de interconexión da la idea de tener conectores en cualquier lugar donde dicha onda llegue, permitiendo entonces a cualquier persona, lógicamente sin necesidad de contar con un cable, a conectarse a ella si posee el código o los conocimientos adecuados para hacerlo. Una de cal, y una de arena, como siempre. Para grandeza de males, todos los días se publican nuevos métodos para romper los sistemas de autenticación de este tipo de redes.

- Red de área local virtual: Como antes, de sus siglas en inglés recibe el nombre de “VLAN” (Virtual Local Area Network). Ocurre que en una misma red física, aún cuando se encuentre en un único edificio, pueden coexistir diferentes redes lógicas. Si éste fuera el caso, tendríamos que multiplicar la infraestructura y por ende el cableado de red por cada una de las diferentes redes que deban

montarse. Para bajar los costos de este tipo de conexiones, se pensó en virtualizar y separar desde un punto de vista lógico diferentes redes aún cuando sean transportadas por el mismo cable. Así nacieron las VLANs. Un uso muy común de las VLANs es la reducción del tráfico de red existente evitando el broadcast desmedido, y segmentando las redes de una determinada forma. Veremos un poco más de esto en futuros apartados sobre VLANs, cómo generarlas, y cómo aprovecharlas para mejorar el rendimiento de nuestras redes.

- Red de área de almacenamiento: Sí, lo adivinaron. En inglés se denominan “SAN” (Storage Area Network). Cuando varios equipos distintos necesitaron conectarse a discos de determinadas características (generalmente asociadas a la robustez, la que como es de esperarse, conlleva que su elevadísimo valor llegue a millones de dólares, a menos que se haga uso de los artículos presentados en números anteriores sobre la forma de armar complejos sistemas de almacenamiento usando ZFS, de mi humilde autoría), se pensó en poner todos los discos en los mismos tipos de sistemas de almacenamiento, y crear una serie de protocolos que permitan que las computadoras que los requerían los “perciban” como si fueran discos locales.

Dadas las elevadísimas velocidades de comunicación que se necesita en estos casos, es muy común ver que este tipo de redes, establecidas entre sistemas informáticos puros y otros de almacenamiento, estén montadas, como mínimo, con fibras ópticas. ISCSI (Internet SCSI) es un ejemplo de protocolo utilizado para interconectar sistemas de almacenamiento con computadoras.

- Red privada virtual: Seguramente habrán escuchado algo sobre las “VPNs”. Esas son las siglas de “red privada virtual” en inglés (Virtual Private Network). Si un sistema, o red, se encuentra lógicamente separada de otra, y posee la capacidad de contactar a la segunda mediante protocolos de red comunes, se puede establecer una interconexión segura entre ellas. Pero esa interconexión, por ejemplo, si se hiciera a través de Internet, permitiría a cualquier persona que tenga un par de manuales leídos (ustedes también lo podrán hacer un poco más adelante en esta serie de artículos) mirar los paquetes de red (para ir acostumbrándonos al lenguaje técnico, este proceso se denomina “sniffing”) que están circulando en algo tan público como lo es, justamente, Internet, y entonces extraer información, en el mejor de los casos, o inyectarla, en los no tan afortunados. Entonces, se desarrollaron estándares y paquetes de software destinados a agregar, a la comunicación a establecer, un grado de criptografía (la criptografía es el arte de escribir con una clave secreta o de un modo enigmático) suficiente para generar una suerte de túnel de una red dentro de otra.

De esta forma, una persona externa a dicha red, que no cuente con los elementos necesarios para poder entender su tráfico verá impedida su malicia, y el usuario que sí lo requiera podrá trabajar conectado a la red de destino tal y como lo haría si estuviera con un cable directamente conectado a ella.

Ese es el concepto subyacente al establecimiento de redes privadas virtuales entre sistemas, muy difundido hoy en día gracias el ancho de banda disponible en una conexión hogareña, y a la simplicidad de implementación de este tipo de redes. El alcance de este tipo de redes es tan variado como las distancias que hay en Internet entre dos sistemas.

Ejemplos de protocolos de VPN son: IPSec (IP Security), SSL-VPN (Secure Sockets Layer Virtual Private Network), o PPTP (Point to Point Tunnel Protocol), cada una de ellas con una implementación en nuestro sistema operativo *nix favorito.

Hay muchos más tipos de redes, siempre que se las diferencie por su alcance, pero con las que enumeramos antes tendremos para divertirnos un buen tiempo, y podremos movernos holgadamente en la mayor parte de los casos que se nos presenten.

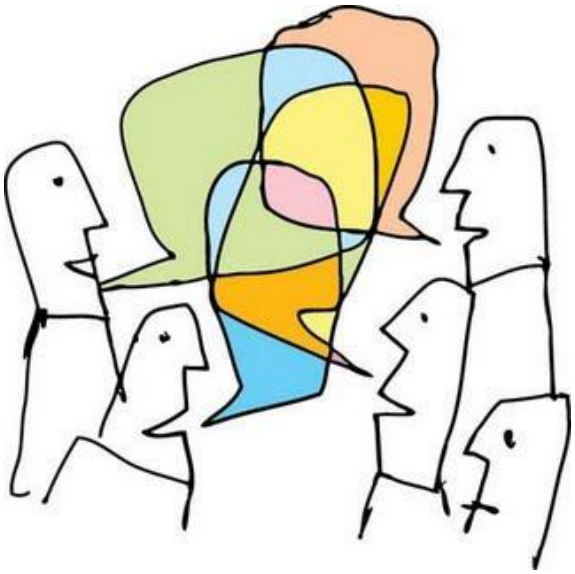
La pregunta para pensar: Tenemos un teléfono celular que se puede conectar a Internet, y a él conectado un auricular Bluetooth. Conectamos el celular a una red WiFi de un cybercafé. Levantamos un programa para usar servicios de voz sobre IP, como lo es Skype. Entonces, ¿a qué tipo de red estaría conectado el auricular Bluetooth? ¿La respuesta? Se la dejo a ustedes, mis queridos Padawanes.

Clasificación por tipo de transmisión

Las redes de datos podrán transmitir en un único sentido, en forma alternada entre el emisor y el receptor, o en forma bidireccional. Es por eso que las comunicaciones, siguiendo este concepto, se pueden clasificar en:

- Simplex: Tal como ocurre con una alarma sonora, que sólo maneja tráfico de información en un sentido (desde el emisor hacia nuestros oídos, generalmente dejándolos bastante doloridos, y alterando nuestros delicados nervios), existen casos donde se utiliza la comunicación unidireccional, y en un único sentido. Un ejemplo podría ser un programa que sólo funcione recibiendo mensajes de un servidor, para luego mostrarlos en una pantalla, y que no envíe ningún tipo de confirmación a dicho sistema.

- Half Duplex: El caso de las comunicaciones Half Duplex es similar al de las Simplex, con la única diferencia de poder cambiar el sentido de la comunicación cuando sea necesario. Primero la comunicación viajará en un sentido, y luego en otro. Casi como si se tratara de una charla entre personas bien educadas. Uno habla y el otro escucha, luego el primero escucha, y el segundo habla. Todo va a ir bien hasta que se establezca el siguiente tipo de comunicación que veremos, que si no está bien orquestado normalmente se verá seguido por una notable escalada de violencia que puede terminar en pugilatos combates, y hasta participación de fuerzas del orden.



- Full Duplex: En este tipo de comunicación, los datos viajarán en ambos sentidos al mismo tiermpo, casi como si se tratara de una autopista de dos manos, permitiendo que las velocidades, y también los errores, sean mucho mayores. Siguiendo el caso de la charla entre personas, veremos que es muy común en estos días encontrar gente que por cuestiones formacionales deficitarias, o sencillamente por una falta de limpieza en sus orificios auditivos no logran percibir que alguien les está hablando y se ponen a hablar sobre la primera voz. Alguien notó tan desagradable hecho, y decidió usarlo a su favor. A ese héroe debemos este tipo de comunicaciones.

Elementos de interconexión

En este apartado veremos algunos de los tantos elementos utilizados en la interconexión de equipos informáticos. Será gracias a estos elementos, algunos más inteligentes, elaborados, y caros que otros, que podremos diseñar un esquema de red acorde a las necesidades que se nos presenten.

Algunos de estos elementos de red son:

- Tarjetas de red: Su sigla es “NIC”, gracias a “Network Interface Card”, su denominacion en inglés. Las tarjetas de red serán aquellos elementos que nos permitirán conectar nuestras computadoras a la red en sí misma. Las tenemos en diferentes sabores y colores: cableadas, de fibra óptica, inalámbricas, y un largo etcétera. Un ejemplo de tarjetas particular es la Ethernet, que conecta, en forma cableada, una computadora a una red. Para hacer esto, posee una dirección física única entre todas las tarjetas de red del mundo, denominada MAC por “Media Access Controller”, constituida por tres octetos de bits expresados en hexadecimal, y otros tres que refieren al número de tarjeta dentro de las generadas por ese fabricante. Si ejecutamos, en nuestras máquinas GNU/Linux el siguiente comando, podremos encontrar la dirección MAC:

```
hecsa@dshecsa01:~$ ip address show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:1f:c6:08:f5:7f brd ff:ff:ff:ff:ff:ff
    inet 10.100.100.2/24 brd 10.100.100.255 scope global eth0
    inet6 fe80::21f:c6ff:fe08:f57f/64 scope link
       valid_lft forever preferred_lft forever
```

Veremos en la entrada correspondiente a eth0 que el campo “link/ether” posee el valor 00:1f:c6:08:f5:7f. Entonces, visitando la página http://standards.ieee.org/develop/regauth/oui/oui.txt podremos encontrar al fabricante de la tarjeta de red:

00-1F-C6 (hex) ASUSTek COMPUTER INC.
001FC6 (base 16) ASUSTek COMPUTER INC.
15,Li-Te Rd.,Peitou,Taipei 112
Taipei 112
TAIWAN, PROVINCE OF CHINA

Entonces, nuestra tarjeta de red habrá sido fabricada por la empresa Asus.

- Repetidores: Un repetidor (“repeater”) es un dispositivo que recibe una determinada señal electrónica codificada en base a una norma preestablecida, la limpia de los posibles errores que pudiera contener, y la reenvía por otro conductor luego de haberla recompuesto, o regenerado.

La idea de un repetidor es la de poder enviar una determinada señal a una distancia mayor que aquella soportada por el medio de enlace que estemos utilizando. Para tener una analogía que nos suene algo más familiar, quienes vivan en lugares donde los canales de televisión deban ser repetidos por emisoras locales, tienen en sus manos un caso de un repetidor.

El canal local toma los programas, les quita la publicidad correspondiente a sus zonas de

origen, luego le agrega las publicidades locales, y retransmite esos programas para ser vistos por los televidentes que viven en las zonas más cercanas.

- HUB: Habiendo entendido lo que un repetidor es, nos preguntamos qué pasaría si intentáramos reenviar la misma señal que hemos recibido no por uno, sino por varios cables (u otro medio) conductores. Cuando hacemos eso, tenemos, a diferencia del caso anterior, un HUB. Como nos podremos imaginar, un HUB es un dispositivo, entonces, que tiene no dos sino varias bocas de red a las que se conectarán diferentes computadoras.

Hoy por hoy, el uso de HUBs está algo dejado de lado, ya que el sólo hecho de retransmitir por todas sus bocas de red el mismo conjunto de paquetes de información si bien ayuda porque interconecta de una forma muy económica dispositivos, también genera un tráfico de red innecesario, por ende degradando la velocidad y prestaciones de dicha red.

Para tener una analogía algo acuosa, vayamos al baño de un restaurant grande, que tenga varios lavatorios conectados al mismo caño proveedor de agua, y abramos varias canillas de una a la vez. La primera tendrá un chorro de agua de buen caudal, pero la segunda canilla bajará dicho caudal de ambas dos. Si abrimos tres el caudal se dividirá por tres, y así consecutivamente, hasta no alcanzar para lavarnos las manos.

Lo bueno de los HUBs es que su valor puede ser casi despreciable para cualquier bolsillo, son bien pequeños, y silenciosos. Otra vez, una de cal y una de arena.

- Switches: Un switch de red puede verse casi como un HUB, con algunas sutiles diferencias que lo transforman en el elemento preferido a la hora de montar una red, inclusive una hogareña, dado su notable descenso de precios en los últimos tiempos. A diferencia del HUB, el switch no reenvía los paquetes de red hacia todas las bocas de red disponibles, sino que lo hace sólo hacia aquella a la que está conectada la dirección MAC que se desea contactar.

Como se podrán imaginar, es notable el descenso de paquetes de red que se envían y que nunca llegarán a un destino útil. Como también estarán pensando, el nivel de lógica que este tipo de dispositivos posee es más inteligente que el de su hermano menor, el HUB.

Tiene en cuenta más datos provenientes de la red, los almacena en alguna tabla interna, y los compara con los valores de los paquetes de red recibidos. La duda que de seguro estarán teniendo es sobre qué tipo de información se intercambia en una red, y si hay varios estratos de información, qué tener en cuenta a la hora de entablar una comunicación entre dos máquinas. ¿No estaban pensando en eso? No hay problema, lo mejor aún está por venir.

- Bridges: Un bridge muchas veces puede confundirse con un switch, al menos en lo que a funcionalidad respecta. La idea de un bridge es la de contar con la inteligencia necesaria para analizar los paquetes de red que llegan en cada una de sus bocas, e interconectar diferentes LANs sin enviar paquetes por todas sus bocas, como lo hacen los HUBs.

Lo interesante de un bridge radica en que al recibir un paquete de red que contiene una determinada dirección MAC, considera que en la boca de red donde lo recibió estará, efectivamente, conectado el equipo que posee dicha dirección, y siempre que reciba un paquete de red cuyo destino sea esa dirección, sólo lo retransmitirá por esa boca, no otra. Nuevamente, vemos una inteligencia superior a la del HUB.

Es común ver bridges cuando se pretende interconectar las redes LAN de dos locaciones geográficamente separadas, utilizando por ejemplo equipos de radio con dos antenas.

El nombre de “bridge” nos da una idea de lo literal que es su analogía con un puente, que une dos partes de una misma ciudad para permitir que autos de un lado pueda llegar al otro.

- Enrutadores: Un enrutador (“router” en inglés) es un dispositivo que permite interconectar redes LAN de DIFERENTES características en lo que a direccionamiento IP se refiere, así como que los paquetes de red provenientes de un lado lleguen en forma adecuada al otro. Si nos detuviéramos en esta parte de la definición pensaríamos en dispositivos similares a los switches o bridges.

El tema es que si bien una máquina puede tener un router predeterminado declarado, también puede tener otros que permitan el envío de paquetes a otras redes. Si éste fuera el caso, encontraría entonces la capacidad de analizar los paquetes de red sabiendo que si deben llegar a una red deben ir en un sentido que he declarado, pero si deben ir en un sentido que no tengo declarado, sencillamente utilizarán el router predeterminado, ya que es él quien sabe para qué lado debe enviar los paquetes de red.

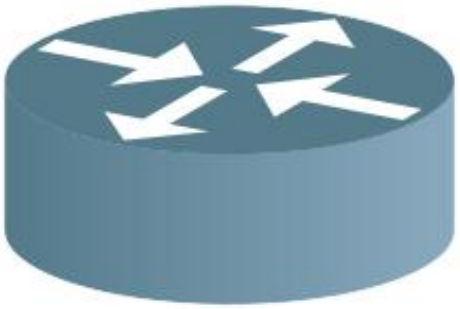
Lo más común es tener declarado un determinado enrutador, y que él se ocupe de manejar los envíos y recepción de todos los paquetes de red, tal como lo hace un controlador aéreo de una torre de control de un aeropuerto.

En nuestras máquinas podemos ver cuál es la tabla que rige las rutas que nuestros paquetes de red tomarán ejecutando el comando:

```
hecsa@dshecsa01:~$ netstat -nr
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS
Window  irtt Iface
0.0.0.0         10.100.100.1    0.0.0.0         UG      0 0
0 eth0
10.100.100.0    0.0.0.0         255.255.255.0   U       0 0
0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U       0 0
0 eth0
```

Observemos por un segundo esta sencilla tabla de rutas. Veremos resaltada la entrada que hace referencia al destino “0.0.0.0”, que simboliza todos los destinos para los que no se haya declarado específicamente una ruta. Eso quiere decir que si quiero enviar un paquete de red a cualquier lugar que no tenga una dirección IP del estilo “10.100.100.x”, lo enviaré a través de la dirección 10.100.100.1. Entonces, el enrutador predeterminado será ese, el 10.100.100.1, que sabe qué hacer con los paquetes de red para que lleguen a destino.

La pregunta que les hago entonces es la siguiente: ¿el enrutador predeterminado que tengo en mi tabla, tiene también declarado un enrutador predeterminado en su tabla? ¿Por qué?



- Cortafuegos: El nombre cortafuegos seguramente no signifique mucho para muchos, pero sí el término inglés “firewall”. La idea del firewall, cuyo nombre deriva de la

industria automovilística (un “firewall” es un separador entre el motor y el habitáculo que impide que un posible incendio nos enrule el flequillo si estamos dentro del rodado) es la de separar diferentes redes permitiendo sólo el paso de determinados paquetes de red, hacia un determinado destino, y desde determinados orígenes. Los hay muy variados, de diferentes caracteristicas, mejores y peores.

Lo cierto es que GNU/Linux, *BSD y OpenIndiana poseen, incorporados en sus núcleos, versiones excelentes de firewalls que nos permitirán dormir en paz al menos en lo que a acceso a los recursos de red refiere. En ellos podremos declarar reglas que funcionan como si fueran derechos de admisión de paquetes de red.

A quienes se lo preguntan, no nos hemos olvidado de los cables, pero eso pertenece a un mundo tan vasto que tendremos que explorarlo en un apartado especial sólo para eso, que a continuación se despliega.

Cableados varios

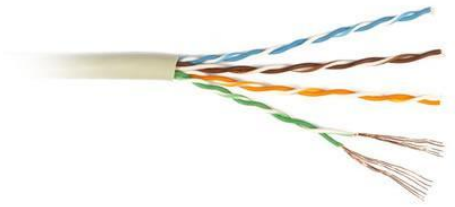
Veremos en este apartado algunos de los métodos más comunes de interconexión de redes de datos utilizando cableados, sea cual fuere su característica o naturaleza.

- Cableado de par trenzado: Este tipo es muy antiguo, pero aún así de los todavia más usados en la industria. Se conoce como “Twisted Pair” por su traducción al inglés. Un par trenzado consta de dos conductores de cobre trenzados uno sobre el otro en forma helicoidal. Se preguntarán el motivo de tan bonito trabajo al hacer una trenza con los conductores, y la respuesta radica en que si analizamos el comportamiento de dos conductores de cobre cuando se colocan en paralelo, estaremos en presencia de una hermosa antena. Y el problema de tener delante una antena es que lo que justamente no queremos es interferencia cuando estemos comunicando sistemas informáticos. Cuando los conductores se trenzan de la forma que lo están en el cable de pares trenzados, cualquier señal por ellos generada es automáticamente cancelada. Las conexiones cableadas de par trenzado pueden desplegarse en distancias de varios kilómetros, luego de los cuales el uso de un repeater se hace necesario si no queremos perder paquetes de red.

El ancho de banda que este tipo de conexión pueda transmitir dependerá en gran medida del largo del conductor, del tipo de aislamiento que posea, y demás factores constructivos.

Según sus características constructivas, podemos encontrar varios tipos de cables de pares trenzados diferentes, como ser los de tipo 3, más antiguos y entonces con soporte para velocidades menores, y los de tipo 5, o tipo 5 especial, que permitirán por ejemplo el uso de redes del tipo gigabit ethernet. Lo que normalmente diferencia a un tipo del otro es la forma en la cual se han trenzado los conductores, que será mucho más densa en el caso de las velocidades mayores.

Muchas veces escucharemos de este tipo de tecnología también como “UTP”, que se corresponde con la sigla en inglés para “Unshielded Twisted Pair”, o par trenzado sin apantallado (se le dice apantallado a una cobertura exterior que un cable posee y que permite eliminar interferencias).



- Cable coaxil: El cable coaxil es algo así como el dinosaurio de los cableados, pero siempre nos sorprende con una nueva aplicación. Está normalmente mejor aislado y construido que el de par trenzado, lo que permite que cubra distancias y velocidades mayores. Típicamente se usan dos tipos de cable coaxil: uno que tiene una impedancia de 50 ohms (el más común, y utilizado en muchas implementaciones) y el de 75 ohms. Los que tengan conexión a Internet por medio del cable de la TV podrán ver que tienen un coaxil completamente funcional.

- Fibras ópticas: La fibra óptica es una clara

muestra de cómo la evolución de las comunicaciones le ganó a la de los procesadores. Mientras que los procesadores crecen en potencia a razón de veinte veces por década, las comunicaciones lo hacen a razón de cientoveinticinco veces, bajando su nivel de error de 0,0001 a aproximadamente cero.

Es en este campo donde juega su papel fundamental la fibra óptica, que no es más que un filamento extremadamente delgado y delicado de fibra de vidrio que transmite haces de luz entre dos extremos. Un transceptor (dispositivo que transforma un medio en otro) convierte los pulsos eléctricos en haces de luz, los que viajan por la fibra a elevadísimas velocidades para impactar en otro transceptor que convierte haces de luz en pulsos eléctricos.

Dado el valor de las redes de fibra óptica, notablemente mayor que las de cobre, es común ver que en edificios donde hay varios pisos que cablear se distribuya la red de piso en piso con fibra óptica, y dentro del mismo piso con cables de cobre, a menos que se trate de interconectar servidores que requieran de elevados usos de la red, en cuyo caso también se hace llegar la fibra óptica hasta ellos.

Las fibras ópticas se pueden clasificar a la vez en dos grandes grupos:

- Multimodo: Las fibras multimodo usan equipamiento de red más barato que las monomodo, y pueden cubrir distancias de por ejemplo unos 550 metros sosteniendo una velocidad de 10 Gbps. El filamento de este tipo de fibra es más grueso que en otros casos, por lo que el equipamiento para darle soporte no debe ser tan preciso, bajando también los costos por ese lado, ya que pueden usar LEDs (diodo emisor de luz, o “Light Emiter Diode”) especiales para emitir el haz de luz.

- Monomodo: Las fibras monomodo son extremadamente delgadas, y por lo tanto también delicadas, y por su pequeño diámetro no soportan el uso de LEDs para emitir haces de luz, requiriendo un laser en su lugar. Este tipo de fibra puede transportar sólo un haz de luz por filamento, cubriendo distancias de 80 Km a una velocidad de 10 Gbps. Es muy común que se usen estas fibras para interconectar sistemas de almacenamiento ubicados en sitios de contingencia, o para armar clústers de alta disponibilidad distribuidos geográficamente.

Rápidas convenciones

Un punto fundamental a tener en cuenta es la velocidad de cada tipo de red. Nos habremos maravillados al ver la velocidad de las redes de fibra óptica, pero es muy posible que no entendamos bien la diferencia entre un caso y el otro.

Para denominar las velocidades de las redes se suelen usar valores basados en la cantidad de “bits por segundo” que ella puede transmitir. A no confundirnos con “bytes por segundo”, ya que el número es notablemente distinto.

Normalmente, los “bits por segundo” se escriben “bps”, y los “bytes por segundo” “B/s”.

Entre las medidas más usadas encontramos las siguientes:

- Kbps: Kilobits por segundo, lo que es mil bits por segundo (1 Kbps = 1000 bps).

- Mbps: Megabits por segundo, lo que es un millón de bits por segundo (1 Mbps = 1000000 bps).

- Gbps: Un gigabit por segundo equivale a mil millones de bits por segundo (1 Gbps = 1000000000 bps).

- Tbps: Aún no tenemos redes que soporten este inmenso tráfico, pero para evitar tener que reescribir este artículo dentro de un par de años, les comento que equivale a un billón de bits por segundo, usando el término latino del billón, y no el norteamericano, referido a “mil millones” (1 Tbps = 1000000000000 bps).

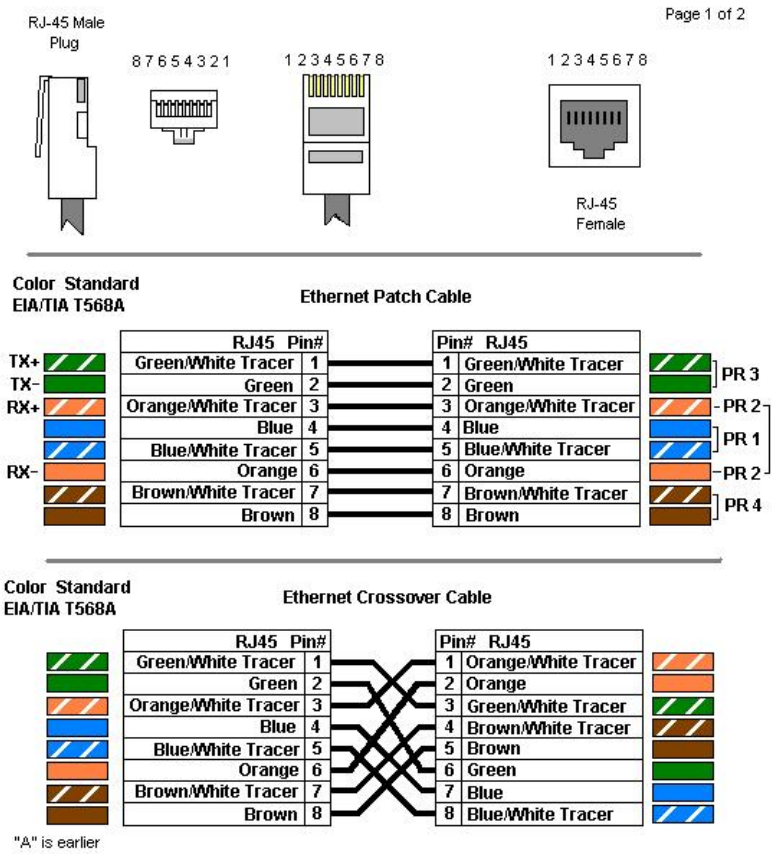
Comparando velocidades

Para que tengamos una rápida idea de las diferentes velocidades de cada tecnología de red, les paso un listín de referencia.

- Red digital de servicios integrados: Conocida por su nombre en inglés como “ISDN” (Integrated Services Digital Network). Su velocidad llega, según se trate de simple o dual, a los 64 ó 128 Kbps. Es un estándar de transmisión de voz, video, datos, y otro tipo de información sobre circuitos tradicionales “PSTN”. PSTN es la sigla de “Public Switched Telephone Network”, o red de telefonía pública conmutada. Sencillamente se usaba la red de teléfonos para transmitir algo más que la voz de un ser humano.

- Tramas T1/DS1: Llega a los 1,54 Mbps. T1 es una especificación de hardware para redes troncales, y un troncal es un canal de transmisión simple entre dos puntos, o dos dispositivos. Uno de esos dispositivos podría ser un teléfono, por ejemplo. Antes de la existencia de las tramas T1, un conductor sólo podía transportar una llamada telefónica, o un flujo de datos, pero con la llegada de la tecnología digital, este tipo de tramas logró hasta 24 llamadas al mismo tiempo, o la suma de sus anchos de banda para comunicaciones de datos.

- Ethernet 10 base T: Llega a los 10 Mbps. Allá en el año 1980 Michael Jackson entrenaba pasos de baile exótico para el video de Thriller, y algunos científicos de la empresa Xerox (sí, la de las fotocopiadoras, y la creadora del entorno de ventanas “X”) sentían sus estómagos crujir porque el invento de unos cinco años antes (1975) ya estaba comercialmente en la calle. Para los que quieran cablear sin caer en la desesperación, les dejo una figura que define de forma bien sencilla los colores de los cables, y la forma de armar un cable que se conecte a un switch (Ethernet Patch Cable), así como de uno que conecte una máquina a otra, sin switches en el medio (Ethernet Crossover Cable):



- Ethernet 100 base T: Ídem, pero con velocidades de 100 Mbps.

- Red óptica sincrónica: Sus siglas son “SONET” y llega a los 155 Mbps (159252,48 Mbps, para ser precisos). Es un estándar de fibras ópticas que permite manejar una jerarquía de señales, entonces definiendo para cada escalafón su velocidad máxima.

- Gigabit Ethernet y Fibra óptica: Llega a 1 Gbps, y vienen en sabor de cobre (Ethernet) y vidriado (fibra óptica). La diferencia entre una y otra tecnología es que la de cobre es más barata, pero también mentirosa, porque el rendimiento de una red de cobre a esas velocidades si logra un 65% del Gbps, podremos sentirnos los reyes del cableado.

- Fibre Channel: Claramente se trata de una fibra óptica cuya velocidad llega a los 2, 4, 8 ó 16 Gbps. Se simboliza como “FC” por su nombre en inglés, y si bien su desarrollo comenzó en 1988 (Guns N' Roses sacaba en ese año el disco “Sweet Child O' Mine”), se estableció como estándar recién en 1994 (Brian Adams sacaba, para delicia de los que nos gustaba abrazar chicas en las fiestas, el tema “Please Forgive Me”, que tantas alegrías me ha dado).

- Infiniband: Esta belleza parte con su norma de los 2 Gbps (Infiniband SDR 1.x), pero llega a los 300 Gbps (Infiniband EDR 12.x), aparte de contar con estrategias de interconexión más integradas al hardware destino que las demás redes. Si hay que armar un clúster de alta performance, es la elegida. Nació de una serie de proyectos del año 1999 (en ese momento yo me estaba rompiendo los cuernos para entender si mis máquinas se transformarían en el anticristo por efectos del año 2000), y en el año 2010 comenzó a ser utilizado para interconectar CPUs.

- 100 GE: Este nombre proviene de "100 Gigabit Ethernet", y si bien anuncia la capacidad de transmitir a esa velocidad, es interesante ver cómo se puede tostar un pollo si decide caminar cerca de un cable de esta tecnología. Consideremos que el mensaje de esta nueva tecnología es que puede sostener dicha velocidad en distancias de 40 Km.

Conclusiones

Espero que para este momento se sientan más cómodos con algunas de las tantas nomenclaturas y tecnologías existentes. Me encantaría volcar más de lo que en mis años de trabajo con diferentes redes de datos he tenido la suerte de utilizar, pero no quiero aburrirlos. Sigan enredándose, nos vemos en el artículo que viene, con mucho más de redes de datos.

Hernán "HeCSa" Saltiel
AOSUG Leader
CaFeLUG Member
Twitter: @hcsaltiel
hsaltiel@gmail.com
http://www.facebook.com/hcsaltiel
http://www.aosug.com.ar

Cuando deja de ser un chiste

POR **CLAUDIO DE BRASI**

Uno de los chistes que hacía en una época acerca de mi muy mal inglés era decir que ese idioma no era bueno para la libertad, al no tener la sutileza de distinguir entre libre y gratis. Dado que no todos somos traductores inglés-español. Se confunde “free” con La Libertad y algo sin costo. Y como los que hacen negocios no pueden hacer mucho con algo que se entrega sin costo... es natural la confusión. Pero algunos usuarios luego de decirlo se quedaban pensando.

Muchos empresarios y empresas siguen creyendo esto. Si fuera cierto, empresas como Red Hat, Suse y Canonical no hubieran progresado. La cuestión que algunas empresas se han hecho mucho más grandes gracias a software cuya filosofía critican. Apple en particular, (Máquinas exclusivas para S.O. exclusivos), que ha tomado a un BSD y aprovechando que la licencia dice que se puede hacer lo que quiera... y lo hizo. Se aprovechó de la licencia y no sólo lo ha hecho exclusivamente comercial. Por allí está DarwinOS, pero no está totalmente libre, Hay elementos que no están disponibles para esa versión del S.O.

Hoy día hay muchos usuarios contentos con GNU/Linux, e incluso hay algunos que no usan nada que no sea completamente libre. Pero incluso esos usuarios a veces caen en la trampa llevados por la inercia del desarrollo, el progreso y la conectividad. Pero la trampa no es sólo en blanco y negro, Libre o no, hay toda una gama de grises. Pero esta trampa no está en los usuarios que hacen un uso intermedio. La trampa está en lo que las empresas ofrecen entre medio y lo hacen con algo que empezó con atisbos de ser libre. Android.

Android tiene el código fuente que cualquiera puede bajar pero está sólo para máquinas exclusivas, la gama de equipos Nexus. Cualquiera que quiera adaptarlo a su equipo con Android debe esperar a que la empresa fabricante del hardware lo libere o dedicarse a hacer una ingeniería inversa para poder adaptarlo. El resultado, muchas ROM mal "cocinadas" o que andan con extraños fallos y hardware que no funciona correctamente, lo que causa un terrible deja-vu de los primeros años del GNU/Linux y el BSD.

Android debería ser tan libre como se pueda. Pero las empresas se han apoderado de él y han privado del código fuente específico para cada equipo y para empeorar las cosas no hay fidelidad al S.O. Siempre esperan y buscan otro, FirefoxOS y UbuntuPhone pueden ser uno de ellos si creen que pueden hacer un mayor negocio con ellos.

La otra cosa que caracteriza a los usuarios es que no exigen los fuentes de las App. Los OpenApp no están tan difundidas como se debería esperar. Algo que llama demasiado la atención, exigen software libre en sus PC pero no en sus tablets y teléfonos.

Si realmente los usuarios de Soft Libre desean seguir siendo libres, deben empezar a exigir que el Soft y el Sistema operativo sean libres para todos los equipos. Si no seremos la inversa Sólo libres en algunos equipos, y eso no es una real libertad. Sino seremos la inversa, en donde nuestra libertad estará sesgada a algunos equipos.

Claudio De Brasi.
@Doldraug.

PD: Cuando un buen chiste ya no hace reír, lo más probable es que el tema se haya vuelto una cosa seria.

Almacenamiento híbrido

POR **JOEL A. CHORNIK**

Almacenamiento híbrido

El concepto de almacenamiento híbrido hace referencia al uso de diferentes tecnologías de almacenamiento de forma combinadas, de esta manera se puede dar una respuesta al dilema de "costo por gigabyte" o "costo por IOPS".

Es decir, a la hora de decidir sobre cómo armar su sistema de almacenamiento de datos, existen dos posibilidades: por un lado, la más común hasta ahora, es aquella en la que uno debe optar por utilizar discos "tradicionales" rotativos –como SATA o SAS–. Pero la nueva forma –que se ha convertido en la vedette del momento en el mercado– es el almacenamiento en discos sólidos o SSD. Evaluemos ambas y veamos sus diferencias para entender por qué terminamos hablando de almacenamiento híbrido.

En primer lugar tenemos los discos SATA/SAS. Éstos funcionan como los viejos discos de pasta, o los CD. Una púa o lector, y un disco –literalmente, algo con forma de disco– que gira. Lo hace a distintas velocidades que son 5400, 7200, 10.000 o 15.000 RPM. Estos números le sonarán familiares ya que la mayoría de los discos de notebook hoy trabajan a 5400 RPM. Es decir 5400 revoluciones en un minuto, lo que equivale a 90 vueltas por segundo. Yendo más en detalle: 11 milésimas de segundo por vuelta.

Entonces, para graficarlo con ejemplos de uso diario, si uno quiere colocar la púa para encontrar un dato, por ejemplo un video, voy a tardar aproximadamente 11ms. En otro caso, si se quieren encontrar 100 archivos diferentes voy a tardar más o menos 1 segundo. Pero, si se está trabajando con datos en Internet y se necesita acceder a cientos de datos por segundo, básicamente un disco rotativo ya no me sirve.

En términos de IOPS (Inputs/Outputs per second) un disco SATA de 5400 RPM va a estar calificado como de 90IOPS, uno de 7200 RPM como de 120IOPS (7200/60), etc. En segundo lugar, ahora disponemos de los discos SSD. Son discos sólidos que no son mecánicos, no tienen nada que se mueva, son completamente electrónicos.

A diferencia de un disco rotativo, un disco SSD de calidad media de hoy en día puede llegar fácilmente a 20.000 IOPS. VEINTE MIL. Quiere decir, que es al menos 200 veces más veloz que un disco SATA para encontrar un dato en una notebook. Y no es difícil encontrar discos de 50 o 100 mil IOPS. Otra ventaja que tienen, es que son más pequeños y consumen mucha menos energía que los discos rotativos. Pero –siempre hay un pero– tienen un gran inconveniente: el costo. Por todas las ventajas que representa y por ser los nuevos en el mercado, estos medios de almacenamiento son caros. Si comparamos este punto con los discos tradicionales, nos encontramos que aproximadamente, el costo por Gigabyte en un disco de 7200RPM puede llegar a estar en los USD 0.03, 3 centavos de dólar, mientras que los discos SSD están más cerca de USD 1 por el mismo espacio. Es decir, tienen un costo 30 veces mayor. Aquí es donde entra mi recomendación del almacenamiento híbrido.
¿Cómo hacemos para lograr un sistema que nos dé la capacidad de discos rotativos, de varios miles de Gigabytes, con la velocidad de los discos sólidos, de varios miles de IOPS por unos pocos cientos de dólares? Y más aún, ¿cómo hacemos para lograrlo sin tener que reescribir todas nuestras aplicaciones? Lo que sea que hagamos debería "verse" para las aplicaciones como si fuera un disco rígido normal.

En los últimos años, diferentes empresas y proyectos han tratado de responder a esta pregunta, y el resultado es que hoy existen algunas opciones:

Discos híbridos: algunas empresas como Seagate han comenzado a vender este tipo de producto que combinan dentro de un mismo disco las dos tecnologías. Básicamente un disco rotativo convencional de 500GB o 1TB, junto a un pequeño

disco sólido de 8GB. Es la controladora del disco rígido la encargada de decidir qué se escribe y qué se lee en cada lugar. Una buena opción para una notebook o pc hogareña que quiera más velocidad sin perder capacidad. (ver http://reviews.cnet.com/8301-3132_7-57572453-98/seagate-ships-3rd-gen-hybrid-drive-now-called-sshd/)

Linux FlashCache: un módulo de kernel de Linux desarrollado y liberado por facebook que permite usar una memoria flash (SSD) como un caché de lecturas o de tipo writethrough. Originalmente hecho para acelerar el uso de MySQL en Facebook, hoy puede aplicarse a un rango más amplio de fines, permitiendo combinar discos rotativos con sólidos como si fueran una sola unidad. (https://wiki.archlinux.org/index.php/Flashcache)

Linux BCache: similar a FlashCache en concepto, es otro módulo con mismos fines que parece estar pronto a ser integrado como parte oficial del kernel (http://bcache.evilpiepirate.org/)

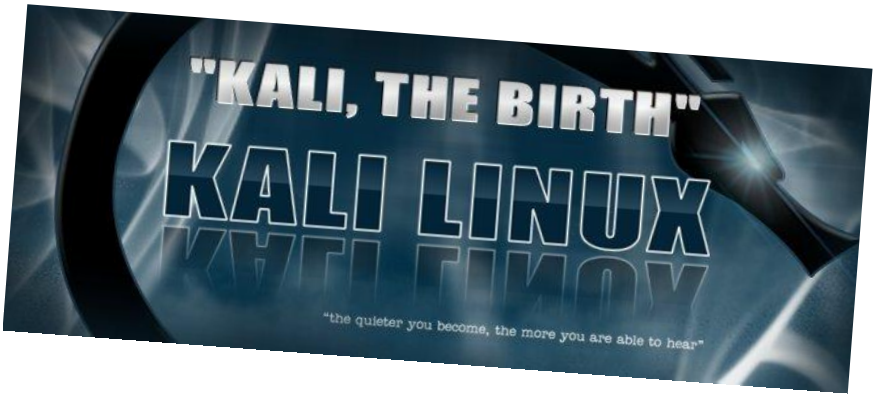
IllumOS (Solaris) L2ARC: una tecnología bien madura que al igual que las opciones de Linux, permite combinar discos como una sola unidad, utilizando los SSD como "Adaptive Replacement Cache" de nivel 2 (L2ARC), siendo nivel 1 la memoria RAM. (http://en.wikipedia.org/wiki/ZFS#ZFS_cache:_ARC_.28L1.29.2C_L2ARC.2C_ZIL)

Windows ReadyBoost: y si, hasta para Windows hay opciones http://en.wikipedia.org/wiki/ReadyBoost

Esta lista es genérica y representa sólo un punto de partida referencial. Seguramente, antes de terminarla, el precio de los discos SSD llegue a ser el mismo que los rotativos, y todo este debate pierda un poco de sentido. Mientras tanto, ¡es bueno saber que podemos acelerar nuestros datos con un poco de software creativo!

Por Joel A. Chornik

Founder, ELSERVER.COM



Kali LINUX El nacimiento

POR **JUAN PABLO LOZANO**

De los creadores de BackTrack llega Kali Linux, la distribución de pruebas de penetración más avanzada y versátil jamás creada. BackTrack ha crecido más allá de sus humildes raíces como un CD Live y ahora se ha convertido en un sistema operativo completo. Kali Linux trae consigo más de 300 herramientas de pentesting y la inclusión de varios programas como Gvim, Leafpad, Arduino IDE, Iceweasel, VLC, Ophcrack, entre otros. Con todo este rumor, es posible que se esté preguntando: ¿Qué hay de nuevo?

Introducción a Kali Linux

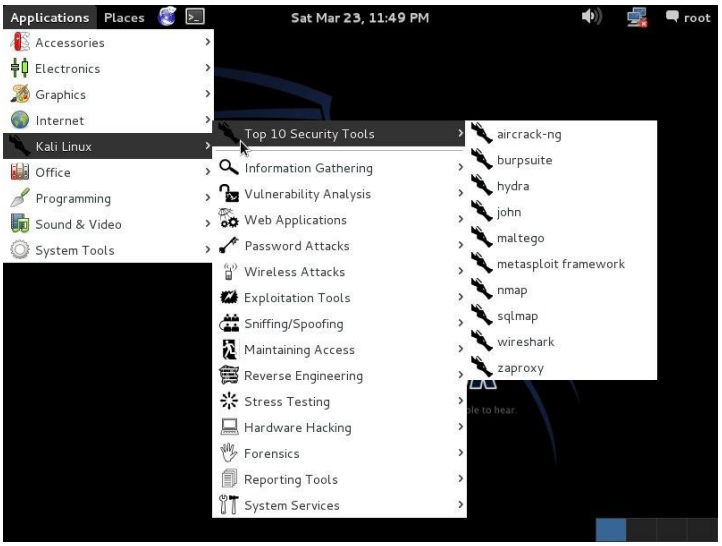
Entonces, ¿cuál es la diferencia entre BackTrack y Kali? Por desgracia no es una pregunta fácil de responder. Es una mezcla entre el "todo" y "no mucho", dependiendo de cómo se utiliza BackTrack.

Desde la perspectiva del usuario final, lo más obvio sería el cambio de la base del sistema a Debian y un sistema estándar de jerarquía de archivos. Esto significa que en lugar de tener que navegar a través del directorio raíz /pentest, ahora se podrá llamar a cualquier herramienta desde cualquier lugar del sistema, ya que cada aplicación se incluye en la ruta del sistema. Sin embargo, hay mucha magia escondida en esa última frase. Rápidamente hay que enumerar algunos de los nuevos beneficios de este movimiento.

- Seguridad y paquete de actualizaciones para Debian: los nuevos repositorios sincronizados con los repositorios de Debian, 4 veces al día, constantemente proporcionan las últimas actualizaciones de paquetes y parches de seguridad disponibles.
- Empaquetado flexible Debian de cada herramienta en Kali: aquí es donde ha pasado la mayor parte del tiempo y esfuerzo. Una lucha implacable contra decenas de empaquetados de herramientas útiles, asegurándose de que los mismos sean compatibles con Debian.
- Largo tiempo de Empaquetado y mantenimiento de las herramientas: muchas de las herramientas en la suite necesitan ser actualizadas (bleeding edge). Esto significa empaquetar y mantener los paquetes para tener la última versión disponible de muchas herramientas que los usuarios necesitan, al día. Lo cual es muy importante.
- Proceso de desarrollo dinámico: a medida que los paquetes fuentes son ahora también compatible con Debian, uno puede fácil y rápidamente conseguir las fuentes necesarias de cada herramienta, modificarlas y reconstruirlas con un par de comandos.
- Compilaciones y personalizaciones de imágenes ISO: uno de los muchos beneficios de nuestro traslado a un sistema compatible con Debian, es la capacidad de construir un medio ISO de Instalación de Kali directamente desde nuestros repositorios. Esto significa que uno puede crear fácilmente sus propias personalizaciones de Kali, así como realizar instalaciones de redes empresariales desde un repositorio local o remoto.

- Desarrollo ARM real: BackTrack 5 trajo consigo un nuevo soporte para hardware ARM. Nuestro constructor ARM fue una versión modificada de la tablet Motorola Xoom, que basta con decir que no duró por mucho tiempo. Para ayudar a solucionar este problema, Offensive Security ha donado un grupo ARM Calxeda a nuestro proyecto, permitiendo un desarrollo fiable y a largo plazo de imágenes Kali para ARM.
- Completa flexibilidad en el entorno de escritorio: nuestros nuevos entornos de compilación y el repositorio, permiten una completa flexibilidad en la generación de sus propios ISOs actualizadas de Kali, con cualquier entorno de escritorio que guste. ¿Prefiere KDE? ¿LXDE? ¿XFCE? ¿Algo más?
- Actualizaciones futuras perfectas entre versiones: otro de los beneficios derivados de la transición a un sistema compatible con Debian es la posibilidad de actualizar perfectamente a una futura versión principal de Kali. Evita tener que volver a instalar en el equipo una nueva versión de Kali. El término correcto es, a lo que muchos llaman "semi-Rolling Release".

Con todos estos cambios (y muchos más), se puede ver la razón del entusiasmo con este lanzamiento.



Punto de vista personal

Al estar basado en Debian, la instalación de Kali Linux no va a traer ningún tipo de complicación, de hecho, pueden usar la guía de instalación de Debian publicada en el número 47 de TuxInfo a partir de la página 22 publicada por Natanael Andrés Garrido.

En el sitio oficial de kali linux http://www.kali.org/ hay numerosas guías para aprender a instalarla, configurarla, y hasta aprender a usar varias herramientas. Te invito a que accedas a los links de interés:

Descarga: si deseas probar o instalar la mejor distribución de pentesting, puedes acceder a la página de descarga. Existen varias versiones tanto para 32 como para 64 bits y ARM
http://www.kali.org/downloads

Documentación: no importa si eres experto o principiante, el sitio y documentación multi-lenguaje podrán informarte o ayudarte en lo que necesites.
http://docs.kali.org/

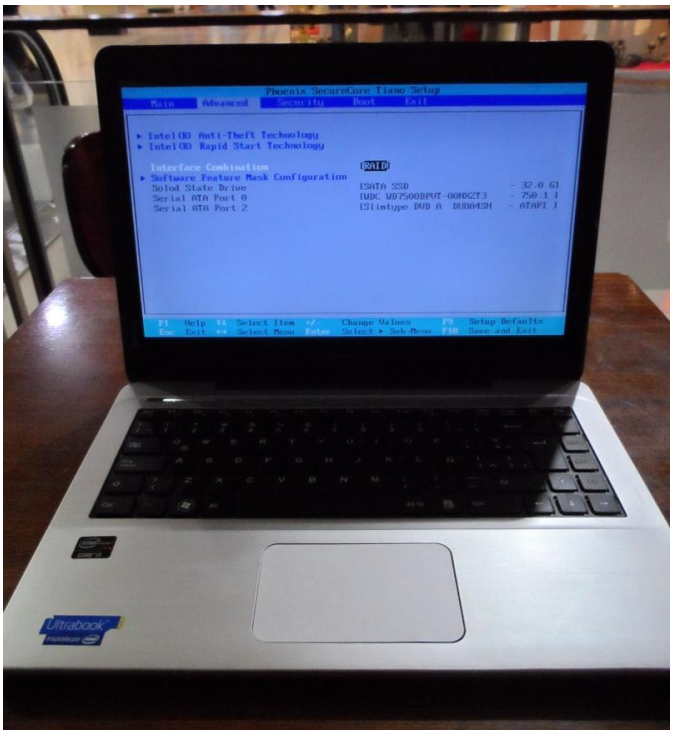
Comunidad: Kali Linux, con su linaje Backtrack, tienen una comunidad muy activa. Iniciamos foros activos, informe de errores y sugerencias de herramientas.
http://www.kali.org/community

Lozano Juan Pablo
lozanotux@gmail.com
twitter: @lozanotux

Probamos una Ultrabook Bangho Zero

POR **ARIEL M. CORGATELLI**

Tuvimos el mes pasado en nuestro laboratorio una Ultrabook de la firma Bangho modelo Zero-B40II1. La misma cumple por supuesto con las condiciones que impuso Intel sobre la categoría Ultrabook. Lo cierto es que la misma vino provista con el sistema operativo Microsoft Windows 8 full. Obviamente desde nuestra tarea la idea era ver cómo se despeñaba con algunas de las distribuciones Linux más simples de instalar.



Para ello procedimos a probarla con dos distribuciones, la primera de ellas la versión LTS de Ubuntu 12.04 en 64 bits y segundo Knoppix 7.5 en 64 bits.
Los resultados fueron excelentes ya que el equipo fue completamente reconocido por las dos distros. Sólo hubo un pequeño contratiempo en el caso de Ubuntu 12.04, en donde no terminaba de reconocer el disco rígido interno, (RAID entre disco SSD de 32GB y disco de plato 750GB) con lo cual no era factible realizar la instalación sin acceder al BIOS de la máquina y modificar algunas funciones. Algo que pasó de forma inadvertida en Knoppix 7.0.5.

Listado del hardware
Pantalla: 14 pulgadas (1366x768) HD LCD LED Backlight
Procesador: Intel® Core™ i5 3317U
Memoria RAM: 8GB DDR3
Disco Rígido: 750GB 5400rpm
Unidad Óptica: CDRW/DVDRW
Lector de Memorias: 6 en 1
Sistema Operativo: Windows® 8 Core
Gráficos: Intel HD Graphics 4000
Wifi: 802.11 b/g/n - Intel® Wireless Display
Bluetooth: Sí
Dimensiones: 344 x 243 × 21 mm
Peso: 1.9 Kg

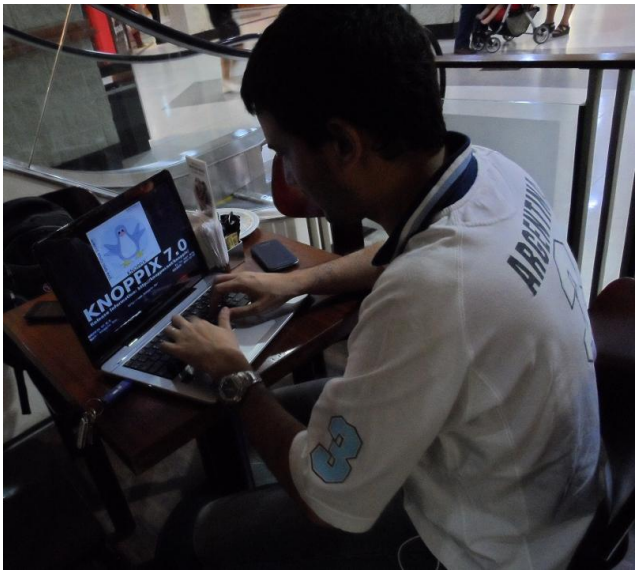
Como habrán visto, el listado del hardware es muy interesante, salvo el arreglo de disco SSD de estado sólido, el cual se utiliza únicamente para levantar la velocidad del equipo y no para almacenar información. Es decir el disco principal en donde vamos a guardar información es el disco clásico de plato de 750gb a 5400 RPM.

Sin entrar en detalles sobre el arreglo de disco, ya que seguramente probándola con una versión más moderna de Ubuntu, ésta lo detectaría sin problemas, decidimos ir hacia una distribución mucho más clásica, con lo cual era el momento ideal para ver qué tan bien se comportaba Knoppix 7.0.5. La sorpresa fue bien grande cuando empezamos a correr la misma desde un PenDrive, donde de primera medida detectaba absolutamente todo, ya sea la placa gráfica tal cual figuraba en la lista de componentes, hasta la placa wifi. Con lo cual nos dispusimos a ejecutar el instalador.

De forma automática arrancó, llegamos al paso crucial en donde nos habíamos quedado en Ubuntu, justamente en Gparted. Ahí mismo la distro detectó

el arreglo completo de los discos, y nos brindaba la opción de poder realizar la instalación. La misma la llevamos sin ningún tipo de problemas, tanto en la operación, como luego el uso del equipo.

Con lo cual teníamos una distribución Linux cargada en una Ultrabook, que originalmente traía instalado un sistema operativo Windows 8 de fábrica, algo que nos parecía raro ya que no tenía en el BIOS la opción del UEFI (algo que hemos hablando ya en números anteriores).



Lo bueno del equipo.
Creo que el primer punto a reconocer de esta ultrabook es la posibilidad de instalación de Linux; esto es algo que siempre intentamos en todos los equipos que recibimos para testear, y que no en todos se puede cumplir. En el caso de la Zero, otro punto a favor es que la misma cuenta con una unidad de CDRW/DVDRW, algo que la hace muy útil para todos los usuarios.
A favor también encontramos que el equipo corre de forma muy fluida, su teclado es más que cómodo, y en lo que respecta a su aspecto físico podemos decir que es muy sobrio y fino.
Por último dentro de sus puntos positivos, tenemos a la BIOS, la cual no tiene UEFI, algo que en muchos equipos con Windows 8 de fábrica ya está implementado. Y con ello no permite la subsistencia física en el mismo disco de dos sistemas operativos. En este caso no hubo problemas ya que no tenía la protección desde el hardware, con lo cual la instalación fue bien simple y no tuvimos que tocar absolutamente nada dentro del BIOS.

Lo malo del equipo.
Arranquemos con que el equipo tiene un arreglo del tipo RAID, no muy óptimo. Por un lado tenemos un disco de estado solido SSD, de 32GB, el cual sólo se usa para el cacheo del disco principal (de plato). El problema radica en que el disco clásico si bien tiene gran espacio para guardar información (750GB) es de 5400 RPM, con lo cual la información que pasa a través del BUS de datos sufre una importante reducción en su velocidad.
Luego otro de los puntos negativos que hemos encontrado es su poca autonomía en lo que respecta a la batería, llegando tan sólo a 3 horas y media a full de carga. Por lo tanto el equipo no cumpliría una exigencia de tiempo de uso clásico de cualquier usuario.

Puntaje
7/10

Compatible con Linux
SI

Enlaces adicionales

Video revisión que le realizamos al equipo
http://www.infosertec.com.ar/blog/?p=42387

Información oficial del producto desde la web de Bangho
http://www.bangho.com.ar/linea_producto/zero/

Conclusión final
En líneas generales el equipo es recomendable para usuarios de Linux, ya que no presenta grandes problemas a la hora de instalar una distro popular. Lógicamente tiene un gran factor negativo, el mismo radica en la autonomía que el equipo nos puede brindar. Algo que hay que tener bien en cuenta a la hora de realizar costosos gastos en equipamiento y que sean lógicamente portátiles. Luego el resto del equipo es completamente apto para el uso de un Linuxero.

Ariel M. Corgatelli
@arielmcorg

Virtualizando con Fedora

una guía básica sobre cómo crear VM's

POR **RINO RONDAN**

En esta guía vamos a mostrarle las herramientas necesarias para poder empezar a crear nuestras máquinas virtuales desde la gráfica.

Lo que necesitamos:

Podríamos utilizar los repositorios que tienen los paquetes más actualizados o dejar el que maneje nuestra distribución, en el caso que queramos usar los más nuevos.

Opcional:
* cd /etc/yum.repos.d/
* wget http://fedorapeople.org/groups/virt/virt-preview/fedora-virt-preview.repo

Chequear si nuestro procesador soporta virtualización:

AMD --> grep svm /proc/cpuinfo

Ejemplo de salida:

```
[root@encarnacion ~]# grep svm /proc/cpuinfo |wc -l
2
[root@encarnacion ~]#
```

En este caso da 2 porque tiene 2 procesadores, por cada uno marca el soporte de svm.

INTEL --> grep vmx /proc/cpuinfo

Ejemplo de salida:

[restauracion@/home/crond1 $] grep vmx /proc/cpuinfo|wc -l
4
[13:26:48]
[restauracion@/home/crond1 $]

En este caso da 4 porque tiene 4 procesadores, por cada uno marca el soporte de vmx

Necesario:

* yum groupinstall “virtualización”

Que contiene lo siguiente:

Group: Virtualization
Group-Id: virtualization
Description: These packages provide a virtualization environment.
Mandatory Packages:
python-virtinst
Default Packages:
libvirt
qemu-kvm
virt-manager
virt-viewer
Optional Packages:
guestfs-browser
libguestfs-tools
python-libguestfs
virt-top

Los paquetes opcionales los vamos a dejar para lo último de esta guía, pero los vamos a ver.

Validación:

Antes debemos instalar este paquete: libvirt-client (si es que no tenemos el binario de abajo)

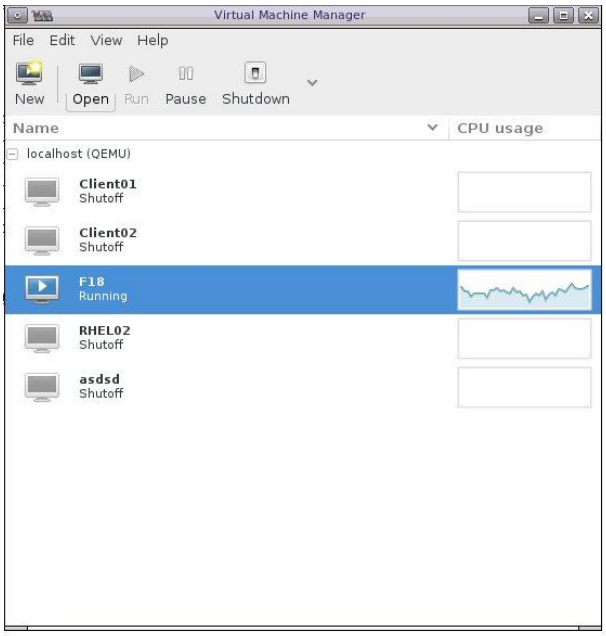
[root@restauracion ~]# virt-host-validate
QEMU: Checking for hardware virtualization

: PASS
QEMU: Checking for device /dev/kvm : PASS
QEMU: Checking for device /dev/vhost-net : PASS
QEMU: Checking for device /dev/net/tun : PASS
LXC: Checking for Linux >= 2.6.26 : PASS
[root@restauracion ~]#

Ahí podemos ver que tenemos todo corriendo como corresponde.

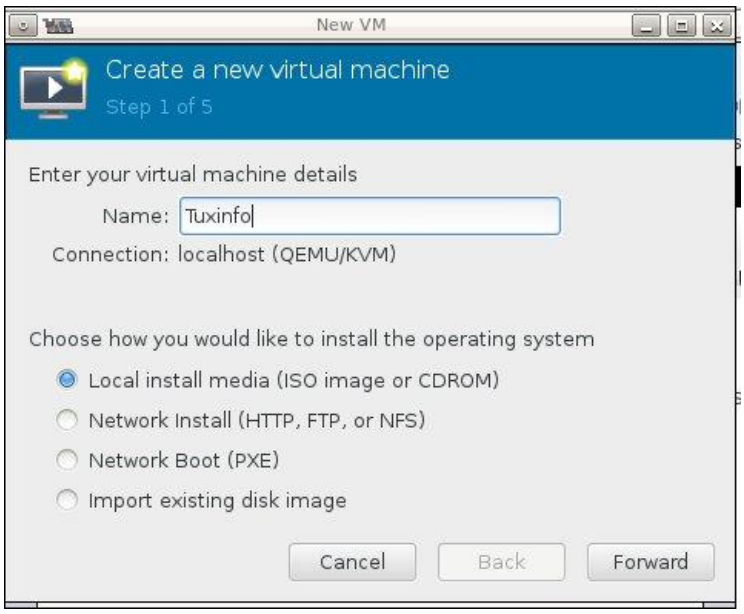
Virt-Manager:

Esta librería libvirt nos permite poder conectar con diferentes hypervisor, en este caso vamos a ver kvm-qemu.

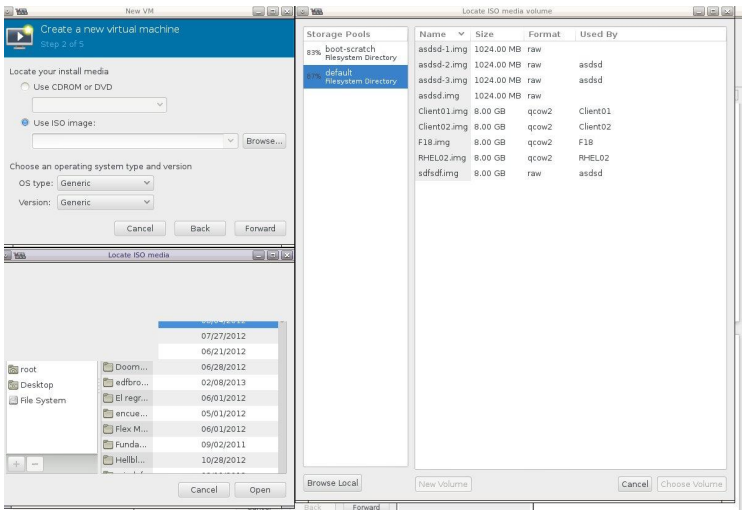


Vamos a ver cómo crear una vm básica sin todavía tener que configurar cuestiones más avanzadas.

Paso 1:
Click en el botón “New”.

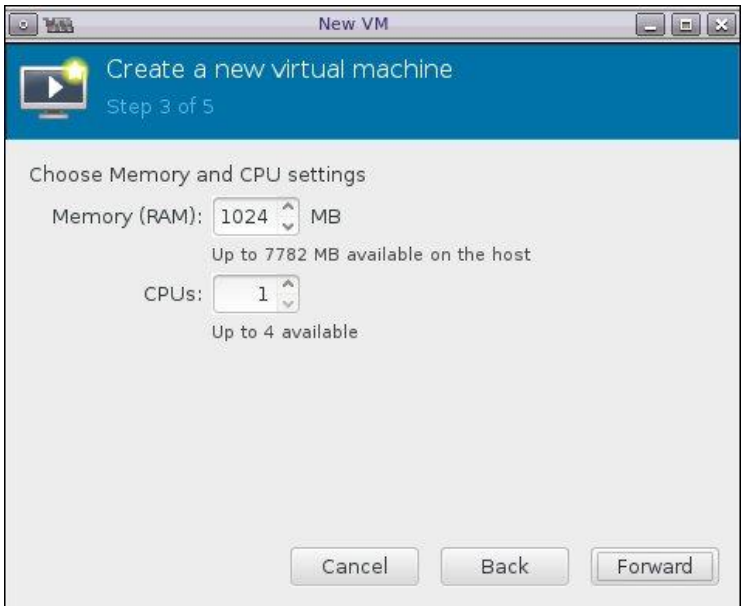


Como se puede apreciar en la imagen, tenemos diversos métodos de instalación, pero en este caso vamos a utilizar el más básico que es instalar desde un CD o alguna imagen que tengamos.

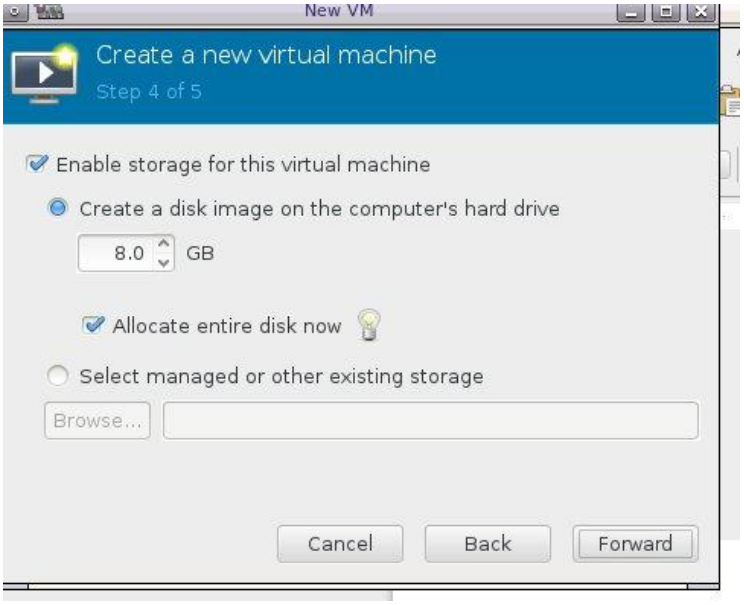


Acá tenemos estas tres imágenes porque primero vamos a tener que elegir desde donde utilizaremos nuestros medios de instalación, en este caso lo vamos a realizar desde una imagen de disco, por eso al hacer click en “browse” se nos abre una ventana con el pool de discos, que todavía no vimos cómo configurar, así que hacemos click en “browse local” para elegir la imagen que necesitemos instalar.

Luego debemos configurar cuanta memoria y procesadores le queremos dar, hay que tener presente que estamos compartiendo recursos importantes (hay algo llamado KSM que nos va a permitir poder compartir espacio de memoria, pero lo dejamos para más adelante).



Ahora sí, cuánto espacio de disco le queremos dar.

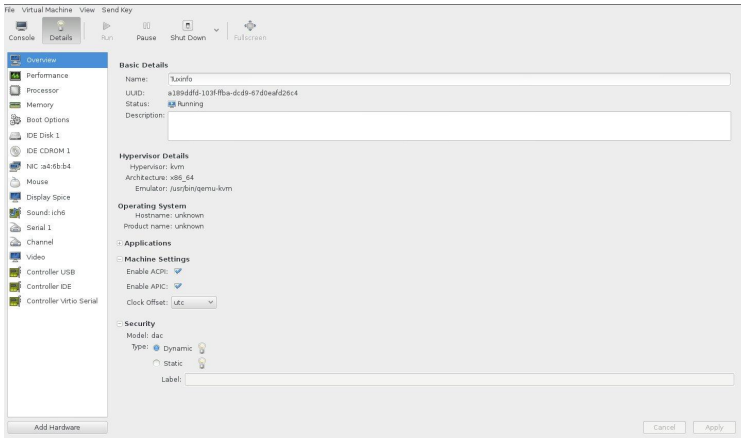


Luego nos devuelve un resumen en donde, si quisiéramos, podríamos poner más opciones avanzadas, pero lo dejamos ahí y finalizamos el "wizard" en donde luego nos va a encender la máquina virtual para que inicie desde el CD que le configuramos.

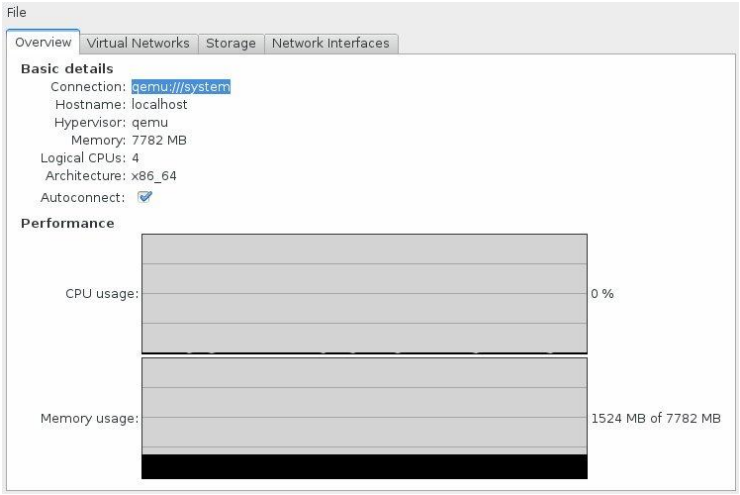
La siguiente pantalla nos muestra el panel de donde vamos a administrar y manejar la consola gráfica de nuestra máquina virtual:

Como verán, tenemos arriba los botones para cada acción, el más avanzado sería "details" porque ahí vamos a poder administrar cuestiones más avanzadas de nuestra máquina virtual, como por ejemplo agregar o quitar hardware.

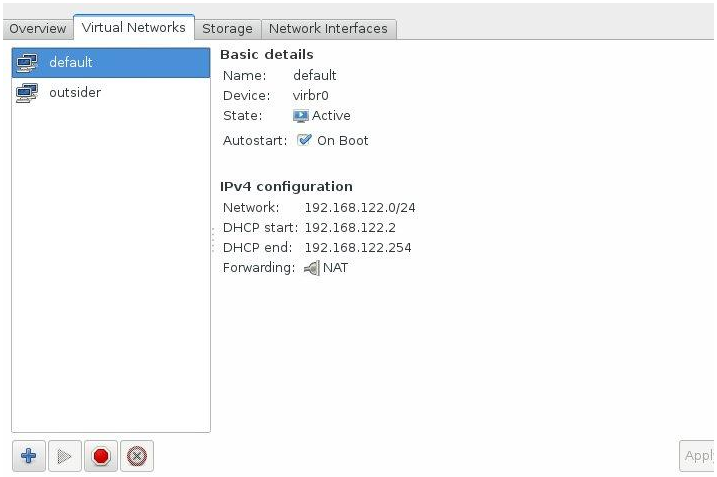
El botón "details" es el que más nos va a interesar dado que podemos configurar varios aspectos.



Algo también importante, que no podemos dejar pasar, son las opciones de conexión en el "virt-manager" (Edit -> Connections Details).



En la solapa “overview” tenemos una vista más general de los recursos de nuestro Host.



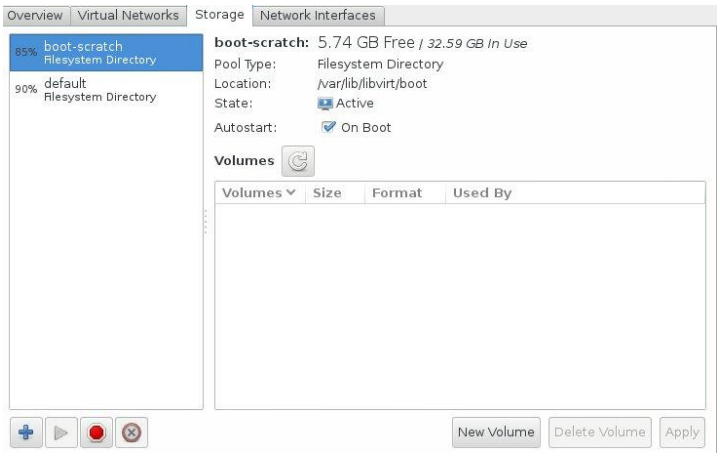
En la solapa “Virtual Networks” tenemos los tipos de redes a configurar, por el momento basta dejarlo así como está. Todas las vm's que generemos van a estar en la red “default” y van a utilizar la interfaz que se crea al instalar el grupo “virtualization”. Donde este tipo de red utiliza un dhcp para otorgar una IP a cada VM que utilice esa red, en este caso en nat, luego en la próxima entrega veremos como utilizarla en Bridge.

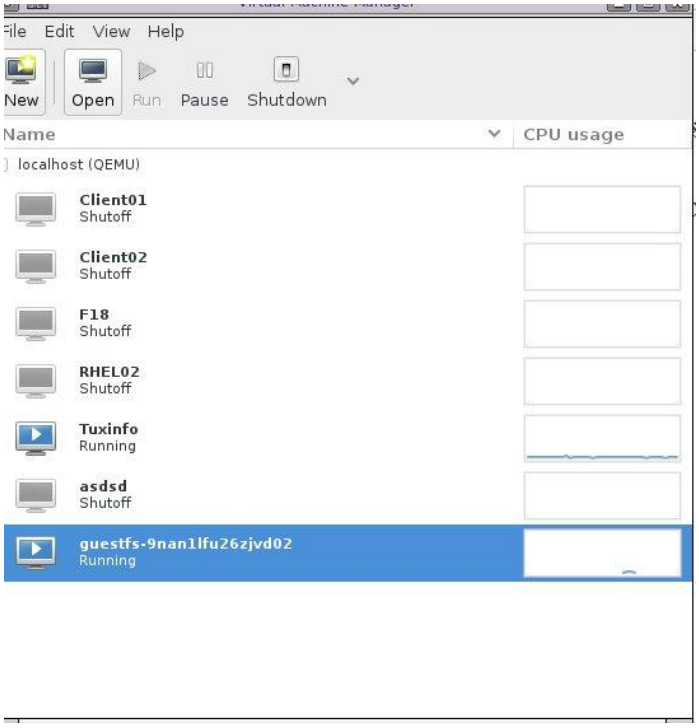
[restauracion@/home/crond1 $] brctl show
bridge name bridge id STP enabled interfaces
virbr0 8000.52540001e5db yes virbr0-nic

vnet0
virbr1 8000.525400b8ebc8 yes virbr1-nic
[15:20:57]
[restauracion@/home/crond1 $]

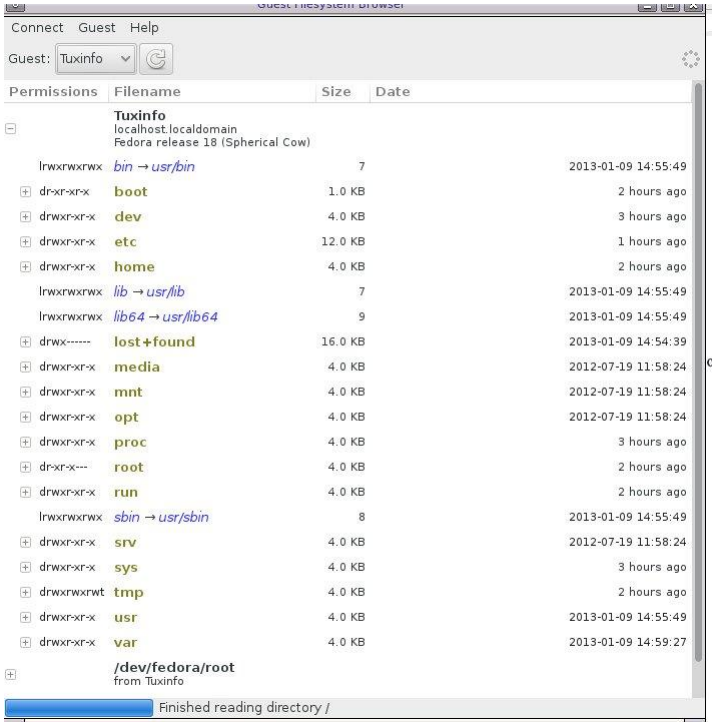
Como verán, con ese comando listo las interfaces de bridge creadas, por ahora como tengo iniciada una sola VM que usa “virbr0” vemos que se activa “vnet0” que es la interfaz de la máquina iniciada.

También tenemos un pool de discos:





Esa última VM que ven como “guestfs” no es nada más ni nada menos, que la conexión que se generó con las herramientas “guestfs-browser”.



Con este programa nos conectamos a algún hypervisor (opción Connect), luego nos lista las VM's y paso posterior elegimos la que necesitamos usar. Esta herramienta actúa como un visor de nuestro sistema de archivo, pudiendo editar los archivos que tenemos. Si la VM está abajo, también la podemos navegar y editar.

Otro programa interesante “Virt-top”.

```
virt-top 15:34:51 - x86_64 4/4CPU 2399MHz 7782MB
6 domains, 1 active, 1 running, 0 sleeping, 0 paused, 5 inactive D:0 O:0 X:0
CPU: 0.2%  Mem: 1024 MB (1024 MB by guests)

   ID S RDRQ WRRQ RXBY TXBY %CPU %MEM    TIME   NAME
    6 R    0    0  104    0  0.2 13.0   7:33.03 Tuxinfo
    -                                           (Client01)
    -                                           (Client02)
    -                                           (F18)
    -                                           (RHEL02)
    -                                           (asdsd)
```

Es una especie “top” pero para nuestras VM's.

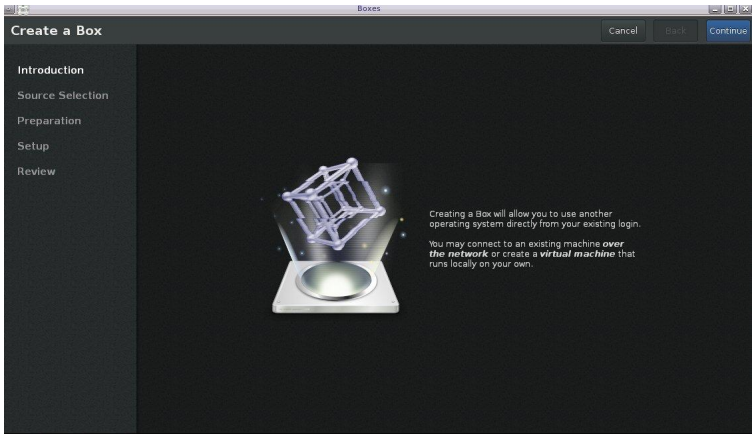
Si vemos los procesos que tenemos en nuestro host, vamos a ver que por cada VM tenemos un proceso.

[root@restauracion ~]# ps -eaf |grep qemu
qemu 8622 1 3 11:29 ? 00:07:34 /usr/bin/qemu-system-x86_64 -machine accel=kvm -name Tuxinfo -S -M pc-i440fx-1.4 -m 1024 -smp 1,sockets=1,cores=1,threads=1 -uuid a189ddfd-103f-ffba-dcd9-67d0eafd26c4 -no-user-config -nodefaults -chardev socket,id=charmonitor,path=/var/lib/libvirt/qemu/Tuxinfo.monitor,server,nowait -mon chardev=charmonitor,id=monitor,mode=control -rtc base=utc -no-shutdown -device piix3-usb-uhci,id=usb,bus=pci.0,addr=0x1.0x2 -device virtio-serial-pci,id=virtio-serial0,bus=pci.0,addr=0x5 -drive file=/var/lib/libvirt/images/Tuxinfo.img,if=none,id=drive-ide0-0-0,format=raw -device ide-hd,bus=ide.0,unit=0,drive=drive-ide0-0-0,id=ide0-0-0,bootindex=1 -drive if=none,id=drive-ide0-1-0,readonly=on,format=raw -device ide-cd,bus=ide.1,unit=0,drive=drive-ide0-1-0,id=ide0-1-0 -netdev tap,fd=24,id=hostnet0 -device rtl8139,netdev=hostnet0,id=net0,mac=52:54:00:a4:6b:b4,bus=pci.0,addr=0x3 -chardev pty,id=charserial0 -device isa-serial,chardev=charserial0,id=serial0 -chardev spicevmc,id=charchannel0,name=vdagent -device virtserialport,bus=virtio-serial0.0,nr=1,chardev=charchannel0,id=channel0,name=com.redhat.spice.0 -spice port=5900,addr=127.0.0.1,disable-ticketing,seamless-migration=on -vga qxl -global qxl-vga.ram_size=67108864 -global qxl-vga.vram_size=67108864 -device intel-hda,id=sound0,bus=pci.0,addr=0x4 -device hda-duplex,id=sound0-codec0,bus=sound0.0,cad=0 -device virtio-balloon-pci,id=balloon0,bus=pci.0,addr=0x6

Todas esas características son los parámetros que se usan para levantar la VM, estas cuestiones avanzadas las veremos en la próxima entrega.

Gnome-Boxes.

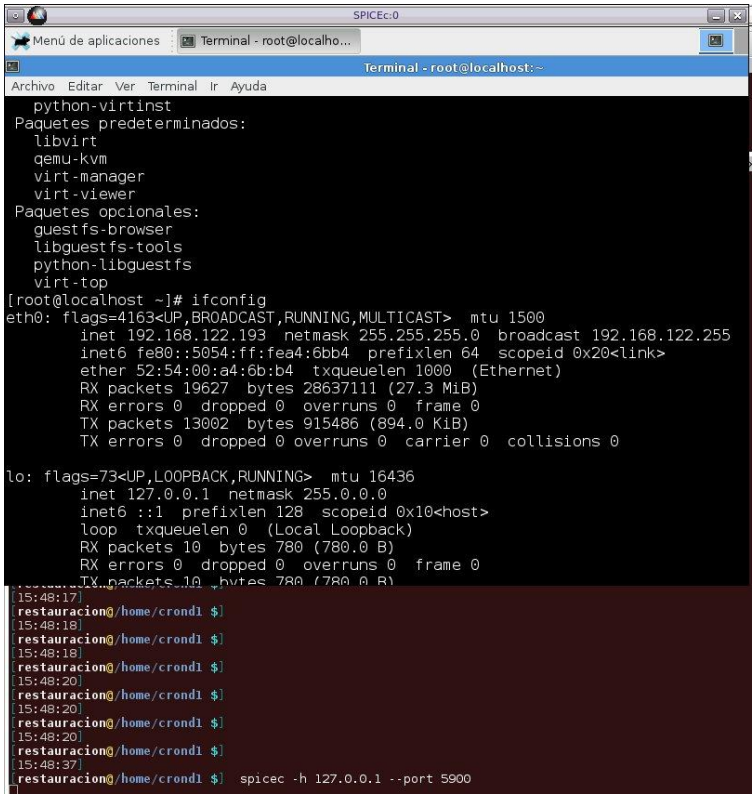
Con esta aplicación vamos a prescindir del “virt-manager” dado que tiene una interfaz mucho más simple para crear VM de manera más rápida y sencilla.



No es tan poderosa como “virt-manager”, pero también nos permite conectar a otros “hypervisor” remotos.

Conectándonos por spice a nuestra VM:

Más información de Spice, la pueden encontrar en Tuxinfo 44
http://www.tuxinfo.com.ar/tuxinfo/?p=621



Este nuevo protocolo nos va a permitir reemplazar a VNC, en donde vamos a tener un rendimiento muy superior pudiendo transmitir video/audio de forma muy eficiente.

Resumiendo:

** Chequeamos que nuestro procesador soporte la virtualización.*
** Instalamos el grupo "Virtualization" para poder contener lo mínimo e indispensable para empezar a virtualizar. También opcionalmente pudimos agregar la rama más nueva de paquetes de virtualización.*
** Creamos nuestra primer máquina virtual con "virt-manager".*
** Editamos y pudimos ver las características de hardware de nuestra VM.*
** Monitoreamos por la consola nuestra máquina virtual.*
** Chequeamos la configuración de nuestro Host (Redes, Storage, recursos disponibles).*
** Utilizamos el protocolo "spice" para conectarnos a la consola.*
** Por medio de "guestfs-browser" pudimos acceder a los datos de nuestra máquina virtual.*
** Pudimos monitorear el uso de cada VM con el comando "virt-top".*
** Con "Gnome-boxes" vimos que tenemos otra forma más fácil de crear nuestras máquinas virtuales sin usar "virt-manager".*

Para la próxima entrega sobre virtualización en Fedora, veremos aspectos más relacionados a la línea de comandos. La idea de esta parte básica era comenzar a crear nuestras máquinas virtuales de una manera más amena y orientada al usuario final. Con el paso de las guías venideras, se irá enfocando más a un ambiente profesional, pero ésto no quita que de esta forma se puedan probar muchísimas cosas.

Rino Rondan
Ambassador Fedora
RHCELPIC2
RHCELPIC2

Tiny Tiny RSS (bye bye Google Reader)

POR **JORGE CACHO HERNANDEZ**

Hace unos pocos días Google anunció que el próximo 1 de julio de 2013 pondrá fin a Google Reader, su servicio para la lectura de feeds RSS [1]

Es bien conocida la forma de actuar de Google que de forma periódica va retirando aquellos servicios que o bien no son rentables o bien no son tan usados como le gustaría o bien no interesan dentro de su estrategia.

Lo que no está claro es el motivo por el que han decidido retirar Google Reader, ya que su uso está muy extendido (es sin lugar a dudas el lector de RSS más popular de internet) y la posibilidad de obtener un rendimiento económico a través de él tampoco hubieran sido complicadas (al igual que ya lo hacen en YouTube con los anuncios).

¿Qué es RSS?

Antes de continuar y por si alguien aún no lo conoce vamos a dar un rápido repaso a este término. Tal y como nos indica la wikipedia: "RSS son las siglas de Really Simple Syndication, un formato XML para sindicar o compartir contenido en la web. Se utiliza para difundir información actualizada frecuentemente a usuarios que se han suscrito a la fuente de contenidos" [2]

O dicho de una forma más llana, RSS nos permite desde un programa o página web, acceder de forma centralizada a todas las actualizaciones de las web a las que nos hemos suscrito, sin que sea necesario por nuestra parte acceder de forma manual a cada una de ellas.

De esta forma en este artículo haremos referencia a "fuente RSS" o "feed" para designar a aquella web que genera datos a través del protocolo RSS.

¿Ha muerto el RSS?

Muchas personas al oír la noticia han vaticinado la inminente muerte del RSS. Google Reader era el mayor servicio de lectura de feeds y muchos han querido ver en esta decisión la intención de Google de matar este formato.

No hay que olvidar que Google Reader además de ser utilizado de forma muy generalizada como lector era el núcleo de toda una gran gama de lectores de feeds cuya principal funcionalidad era que obtenían de Google Reader el listado de feeds o se sincronizaban con él. Google Reader era centro neuronal de una gran red de servicios y programas de lectura de feeds, red de donde Google no obtenía beneficios, pero favorecía que otras empresa sí lo hicieran a través de ella.

Puede que efectivamente ésta haya sido la principal causa para cerrar Google Reader, pero intuyo que Google no ha querido matar el RSS, sino tan solo la forma de consumirlo en la actualidad.

Habrá que estar atento en el medio plazo a la estrategia de Google porque tal vez quieran reforzar el uso de apps como Google Currents (donde cualquier fuente que genere RSS puede darse de alta y crear una publicación) o tal vez Google+, que sigue sin tener la presencia que a Google le gustaría y sobre la que están empezando a centralizar todos los servicios que ahora tenían dispersos (Picassa, etc …)

¿Y ahora qué?

Tras el anuncio de la desaparición de Google Reader hace falta buscar alternativas para poder seguir leyendo nuestras fuentes RSS.

Alternativas en la red hay muchas y en muchos casos buenas, pero casi todas ellas tienen el mismo peligro que Google Reader, no son de código abierto. Es decir, el día en que sus desarrolladores o las empresas que las mantienen dejen de hacerlo nos encontraremos de nuevo en el punto de partida.

En el caso del Reader hemos tenido la suerte de que Google siempre ha habilitado un método de exportación de nuestras fuentes RSS (los famosos ficheros OPML y recientemente a través de Google Takeout), pero hoy en día existen servicios web que nos permiten la importación de nuestros datos pero no su exportación. Esto nos crea una dependencia nada deseable, y por supuesto impide cualquier migración de nuestros datos a otra plataforma.

Urge por tanto encontrar una solución libre que nos permita disfrutar de nuestras fuentes RSS y al mismo tiempo que no nos quite la libertad de gestionar, importar y exportar nuestros datos como deseemos. El hecho de que sea libre permitirá además que si un día el desarrollador que la mantiene dejara de hacerlo cualquier otra persona de la comunidad podrá continuar su trabajo.

Tiny Tiny RSS

En nuestro caso la mejor alternativa que hemos encontrado se llama Tiny Tiny RSS, un servicio que deberemos instalar en nuestro propio servidor y que no sólo cubre al 100% la funcionalidad de Google Reader sino que la mejora en muchos aspectos y además es software libre.

Muchos de vosotros ya tendréis un servidor -propio o virtual- donde tenéis alojadas vuestras webs, vuestro owncloud, vuestra wiki, etc … por lo que instalar Tiny Tiny RSS no tendrá mayor dificultad. Los que aún no tengáis un servidor no dudéis en animaros ya que no hablamos de grandes máquinas ni de costosas inversiones. Por menos de 10 € al mes podemos disfrutar de un servidor virtual de Digital Ocean [3] o de Amazon [4], o por muy poco dinero comprar uno de tantos micro-ordenadores que están saliendo al mercado (como la Raspberry Pi [5]) que perfectamente pueden alojar Tiny Tiny RSS.

Para instalar Tiny Tiny RSS lo primero que deberemos hacer es ir a su web [6] y dentro de la pestaña “Wiki” ir al final de la página y copiar el código de descarga del archivo comprimido (a la hora de escribir este artículo estaba disponible para descarga la versión 1.7.4 con fecha 15 de marzo).

Los pasos que describiremos a continuación están basados en la guía de instalación que nos proporciona la propia web de Tiny Tiny RSS [7]

Iremos a nuestro servidor y nos posicionaremos dentro de la carpeta pública de Apache (normalmente suele ser /var/www/ aunque se puede cambiar a cualquier otra). Descargaremos el fichero con el código fuente, lo descomprimiremos y borraremos el archivo comprimido:

wget https://github.com/gothfox/Tiny-Tiny-RSS/archive/1.7.4.tar.gz
tar -xzvf 1.7.4.tar.gz
rm 1.7.4.tar.gz

A continuación tendremos que conectarnos a nuestro servidor de base de datos (que podrá ser mysql o pgsql) y crear una nueva base de datos para almacenar todas las tablas de Tiny Tiny RSS. Para ejecutar este paso y el siguiente podremos hacerlo usando comandos desde la línea de comandos o con utilidades gráficas del tipo phpMyAdmin. Una vez hecho esto ejecutaremos dentro de esa nueva base de datos el script correspondiente (bien schema/ttrss_schema_mysql.sql o bien schema/trss_schema_pgsql.sql).

A continuación entraremos dentro de nuestro servidor al directorio donde se nos ha instalado Tiny Tiny RSS, renombraremos el fichero config.php-dist a config.php y lo editaremos para poner los parámetros correspondientes a nuestro servidor:

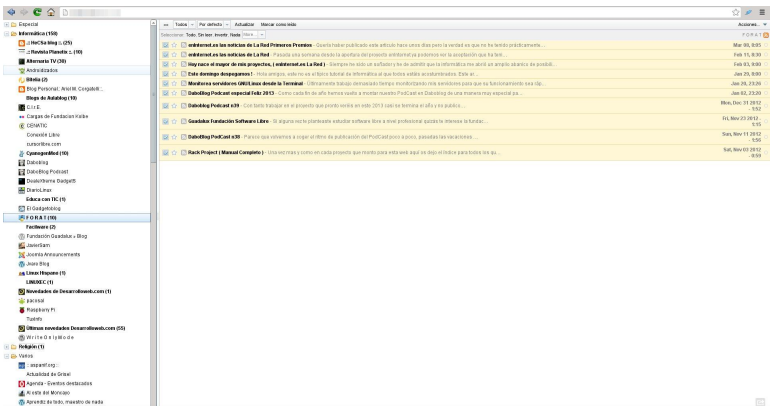
cd Tiny-Tiny-RSS-1.7.4
mv config.php-dist config.php
vi config.php

En mi caso fue necesario modificar los siguientes parámetros: DB_TYPE para asignarlo a "mysql", DB_HOST para asignarlo a "localhost", DB_USER para escribir mi usuario de base datos, DB_NAME para poner el nombre de la nueva base de datos, DB_PASS para escribir la contraseña de la base de datos y SELF_URL_PATH para poner la dirección web que usaremos para acceder a nuestro servicio de Tiny Tiny RSS.

Para evitar problemas de permisos vamos a asignar como propietario de toda la instalación al usuario que usa Apache y a continuación abriremos los permisos de cuatro directorios (si no lo hacemos nos lo solicitará Tiny Tiny RSS en la primera ejecución)

chown www-data:www-data -R ../Tiny-Tiny-RSS-1.7.4
chmod -R 777 cache/images
chmod -R 777 cache/export
chmod -R 777 feed-icons
chmod -R 777 lock

Con esto ya tenemos Tiny Tiny RSS instalado y ya podremos acceder a él desde nuestro navegador con los datos usuales: admin / password (no te olvides cambiarlos la primera vez que accedas)



Si has llegado hasta aquí no creo que haga falta que te indique cómo tienes que importar tus feeds o cómo crear nuevas fuentes. Arriba a la derecha de la pantalla, bajo el menú de “Acciones” y la opción “Preferencias” tienes todo lo que necesitas para configurar tu Tiny Tiny RSS (también encontrarás opciones para configurar diferentes plugins de Twitter, Google+, etc.

Pero aún nos queda un paso, tal vez el más importante. Hemos instalado Tiny Tiny RSS y hemos importado nuestros feeds, pero tenemos que configurar el servidor para que estas fuentes se actualicen de forma automática.

Dependiendo del nivel de acceso que tengamos a nuestro servidor podremos hacerlo configurando un demonio, programando una tarea cron o realizando una configuración para que Tiny Tiny RSS se actualice sólo cuando accedamos a él a través de nuestro navegador.

Si tenemos acceso de root a nuestro servidor lo más cómodo es configurar un demonio. Para eso tenemos que ejecutar el comando php ./update.php -daemon. Es muy importante tener en cuenta que Tiny Tiny RSS nos indica que este proceso no hay que lanzarlo con el usuario root. Además para evitar que el comando se ejecute en segundo plano y que no se cierre cuando cerremos la ventana desde la que hemos lanzado el comando añadiremos el prefijo nohup y el sufijo &

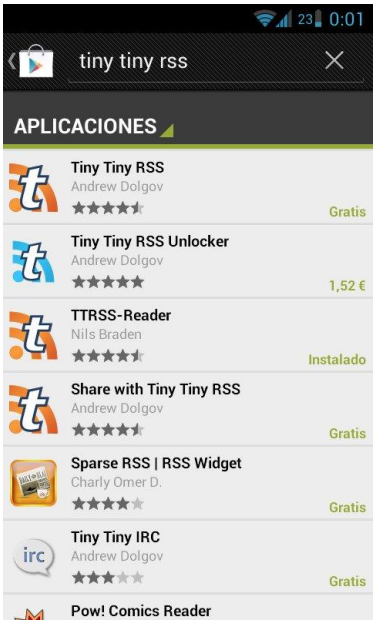
nohup php ./update.php -daemon &

Para más información sobre cómo configurar el demonio o sobre formas alternativas para actualizar los feeds puedes acceder a la información que nos proporciona Tiny Tiny RSS en su web [8]

Ahora sí, ya tenemos todo instalado y bien configurado para leer y disfrutar de nuestros RSS.

¿Y desde mi móvil y tablet?

¡Por supuesto! También puedes acceder a tu servidor de RSS desde tu móvil o tablet Android. En Google Play encontrarás no sólo una sino dos apps para hacer esto.



Una de ellas (la creada por Andrew Dolgov) es la oficial del proyecto. Como veis hay una versión gratuita, que es 100% funcional durante una semana. Pasado dicho tiempo tendremos que comprar la versión de pago por 1,52 € (pago único). La otra alternativa es usar la aplicación creada por Nils Braden que como vemos es gratuita. Ambas aplicaciones son igual de buenas y de rápidas (incluso visualmente son casi idénticas). En nuestras manos quedará elegir si queremos pagar ese pequeño dinero (que imagino servirá para colaborar en el desarrollo de Tiny Tiny RSS) o usar la aplicación gratuita.

Ahora sí, ya podemos decir bien alto "Bye Bye Google Reader". Gracias a Tiny Tiny RSS no sólo tenemos una aplicación con más y mejores funciones que la de Google, sino que además tenemos un cliente móvil más rápido y mejor diseñado. Y por último, y precisamente por eso más importante, tenemos una aplicación 100% libre, con acceso a su código fuente, lo que siempre será una garantía de que nuestros datos están seguros y que el proyecto siempre podrá ser continuado si el desarrollador actual decide pararlo.

Jorge Cacho Hernández
https://about.me/jorge.cacho.h

PD: Al parecer también es posible instalar Tiny Tiny RSS desde los repositorios SID de Debian [9], en mi caso no lo he hecho así por tratarse de un servidor estable en producción, pero no dudes en comentarme si te ha funcionado este método directo.

Sirva este artículo como homenaje y agradecimiento a Aaron Swartz, co-creador del protocolo RSS y activista de la liberación de la información. Tras liberar miles de sentencias federales de EEUU y hacer lo mismo con otros miles de artículos académicos del MIT se vio envuelto en una serie procesos judiciales. En el momento de su muerte se enfrentaba a más de 4 millones de dólares en multas y a hasta 50 años de cárcel. El 11 de enero de 2013, con 26 años de edad, Aaron Swartz decidió quitarse la vida. Tal y como escribió Tim Berners-Lee (creador de la web): "hemos perdido a uno de los nuestros" [10]

Enlaces

[1] http://googleblog.blogspot.com.es/2013/03/a-second-spring-of-cleaning.html
[2] http://es.wikipedia.org/wiki/RSS
[3] https://www.digitalocean.com/
[4] http://aws.amazon.com/es/
[5] http://www.raspberrypi.org/
[6] http://tt-rss.org
[7] http://tt-rss.org/redmine/projects/tt-rss/wiki/InstallationNotes
[8] http://tt-rss.org/redmine/projects/tt-rss/wiki/UpdatingFeeds
[9] http://packages.debian.org/sid/all/tt-rss/download
[10] lists.w3.org/Archives/Public/www-tag/2013Jan/0017.html

M4. Mobile Malware for Mobile Money

POR **JORGE MIERES**

Hablar de códigos maliciosos (o malware) asociado al robo de dinero, quizás ya no genera tanto asombro con tantos ataques informáticos que estamos viviendo. Sin embargo, hace diez años atrás, sí lo era ¡Quién lo hubiera imaginado!

Sin lugar a dudas el malware ha evolucionado a tal punto que en la actualidad existe una gama realmente amplia de estos programas que en rigor se encuentran diseñados para generar algún tipo de daño en nuestros equipos, pero, fundamentalmente… en nuestros bolsillos.

Esta natural evolución siempre va acompañada con la tecnología y bajo esa perspectiva, los ataques informáticos se diversifican, permitiendo a los delincuentes informáticos ampliar el abanico de cobertura para sus campañas de ataques. Donde bajo este escenario, los dispositivos móviles del tipo Smartphone poseen un especial sabor de cara a las actividades delictivas.

¿De qué estamos hablando? Hablamos de que con el Smartphone en mano, básicamente podemos hacer cualquiera de las cosas que cotidianamente hacemos a través de la notebook o el equipo de escritorio. Incluso, transacciones bancarias.

Entonces es lógico pensar que los delincuentes informáticos intenten explotar las debilidades humanas y tecnológicas. Humanas, recurriendo a las clásicas maniobras de Ingeniería Social; y tecnológicas a través de códigos maliciosos diseñados específicamente para las plataformas que soportan estos dispositivos. Siendo sin lugar a dudas Android lo más atacado debido a su condición de “más popular” a nivel mundial.

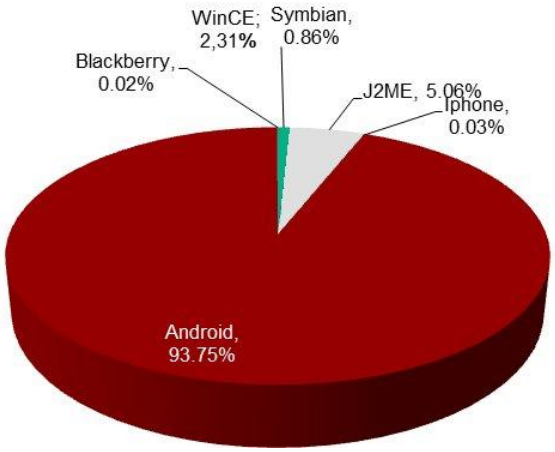
Según el Google Press Summit celebrado a principios del año pasado, para ese entonces la inyección en torno al uso de telefonía celular en América Latina alcanzaba los 600.000.000 de dispositivos, de los cuales 200.000.000 eran Smartphone (aproximadamente un dispositivo y medio por cada habitante). Seguramente en la actualidad la balanza ha cambiado y me atrevo a decir que esos números probablemente hayan sufrido un efecto reverso en la región estimando que existen más Smartphone que celulares genéricos.

Y sí, son muchos dispositivos. Pero la pregunta de rigor en materia de seguridad es qué tan protegidos están, ya que los ataques convencionales de PCs también están a la orden del día en los Smartphone: ataques de phishing, propagación de códigos maliciosos, campañas de scam, robo de datos, robo de identidad, espionaje, sabotaje, entre muchos otros que también incluyen delitos más complejos como la captación de menores por parte de pedófilos.

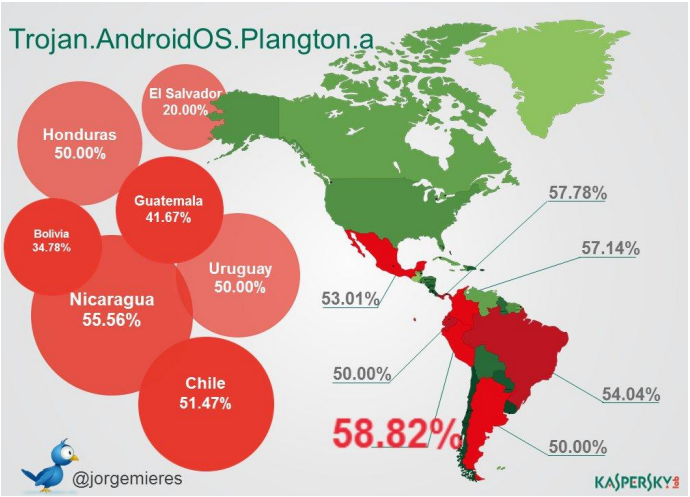
Estimados amigos, hagan la prueba ustedes mismos y analicen la siguiente simple pregunta sobre uno de los conceptos de prevención más antiguos, clásicos y repetitivos en torno a consejos de buenas prácticas de seguridad: ¿Cuántos usuarios de Smartphone bloquean el dispositivo con contraseña? ¿Vamos entrando en situación?

Los códigos maliciosos para Android son una realidad muy crítica. Para tomar un poco más de conciencia respecto a este punto en particular, sólo durante el año 2011 se han desarrollado más códigos maliciosos para Android que durante el rango que va desde el año 2004 al 2010. Lo que significó un aumento en la tasa de propagación de un 600% durante el año en cuestión. Ascendiendo además un 300% más sólo en el primer semestre de 2012. Claro que sigue en aumento.

El siguiente gráfico muestra el porcentaje de malware diseñado para las diferentes plataformas empleadas en Smartphone (fuente: Kaspersky Lab):



Como podemos observar, más del 90% del malware diseñado para móviles está orientado a Android ¿Es una casualidad? Claro que no. Pero… ¿cuáles son los códigos maliciosos con mayor relevancia en América Latina?



El top tres está liderado por un código malicioso que se propaga a través de juegos, con más el 50% de tasa de infección. Las aplicaciones de terceros y especialmente los juegos constituyen un clásico vector de propagación que debe su mayor tasa de éxito a la falta de concientización por parte de los usuarios que, además, no leen los requisitos que solicita determinado juego, por ejemplo, acceder a sus conexiones telefónicas y contactos ¿para qué querría esto un juego?, en realidad la pregunta debería ser ¿para qué quiere esta información un malware? Para enviar mensajes de texto corto de forma clandestina y así acumular dinero en la cuenta de los atacantes. El nombre genérico de este malware es Plangton.

La siguiente imagen muestra la distribución geográfica de Plangton y su impacto en América Latina:

El segundo lugar lo posee un malware conocido como Lotoor ("root all" si lo lee de derecha a izquierda). Lotoor no es más que un código exploit que se utiliza para eliminar las restricciones que posee por omisión la configuración del sistema operativo. Básicamente realiza un rooting mediante el cual el atacante logra tener acceso completo al dispositivo.

Por último, el tercer lugar y con menos del 5%, se encuentra el llamado de forma genérica como FakeInst. Un malware en completo crecimiento que centra su maniobra de ataque en el envío clandestino de mensajes de texto corto (SMS) y que habitualmente llega a infectar los equipos simulando ser una aplicación de interés.

A description...

Como mencioné líneas arriba, el malware diseñado para el envío de mensajes SMS está tomando mucha relevancia en América Latina debido a que operan a través de sistemas de afiliados con números de todo el mundo. Pero esto, será materia de discusión de una futura entrega donde además explicaremos cómo pueden ser utilizados de forma maliciosa los smarphone "rooteados" y cómo transforman tu equipo (cuando no está protegido) en un completo "zombi" reclutado por alguna botnet.

Jorge Mieres, Analista de Malware del Equipo Global de Análisis e Investigación de Kaspersky Lab.Microsoft MVP Enterprise Security y fundador de MalwareIntelligence - @jorgemieres

Myplex música en línea multiplataforma

POR **SAMUEL MORALES CAMBRON**

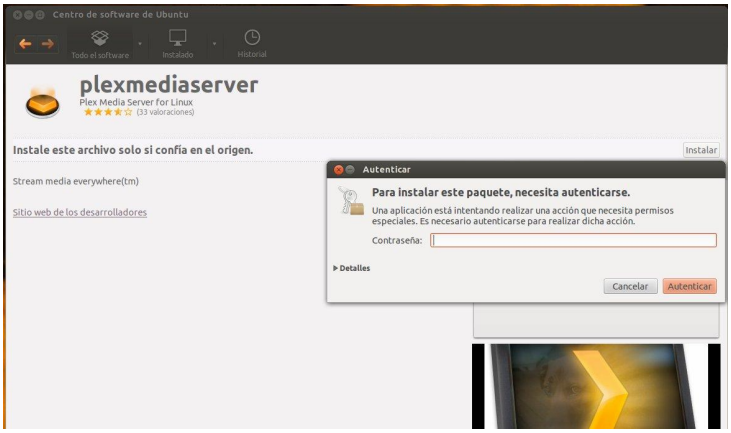
Recuerdo aquella primera vez que vi XBMC, un software maravilloso que convertía tu PC en un poderoso centro multimedia, la verdad quedé encantado e incluso me animé a montarlo en mi casa y convertirlo en nuestro reproductor de películas y música favorito.

Luego leí sobre otro interesante software que me llamó la atención al igual que en su momento XBMC, así que voy a enfocar este breve artículo en una de sus bondades, la de almacenar música y poder reproducirla desde cualquier dispositivo llámese tableta, smartphone, pero ojo que puede hacer muchas cosas más .. Y si te encanta la música como a mí, entonces quizás esto es para ti ...

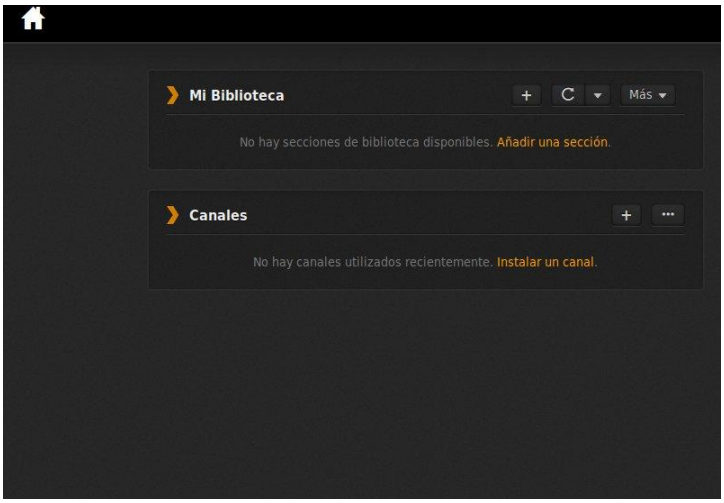
MyPlex

Lo primero que haremos será dirigirnos a la página oficial del proyecto y en el menú "Plex on Your Desktop" descargamos el archivo de instalación correspondiente a nuestro sistema operativo, en mi casa bajo el .deb correspondiente al sistema Ubuntu http://www.plexapp.com/getplex/

Una vez descargado el archivo DEB sólo hacemos un doble click para su instalación, confirmamos instalación escribiendo nuestra contraseña y listo, para más comodidad lo agregamos al dock de nuestra distribución.

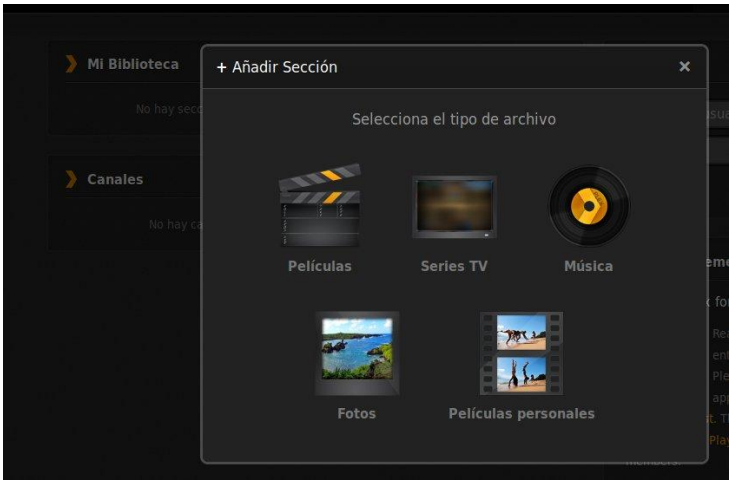


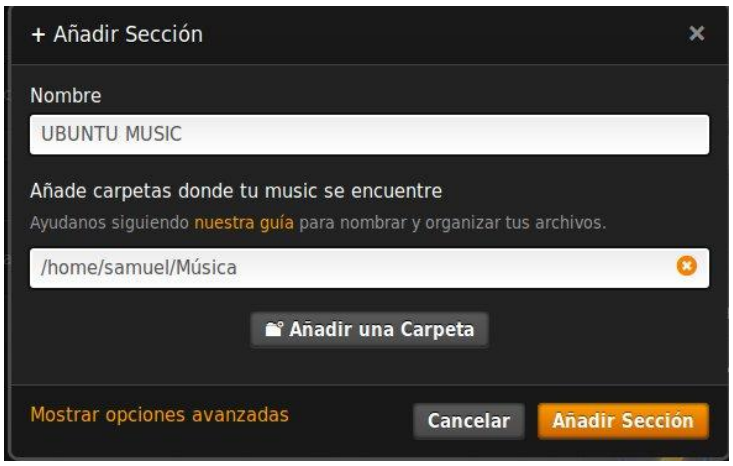
Una vez ejecutado el programa nos va a enviar directo al navegador, el cual será nuestra fuente para agregar más contenido multimedia, aquí está la parte interesante, lo primero será entrar al menú "Mi biblioteca" y "Añadir una sección" ahora en la ventana siguiente vamos a dar la ruta donde tenemos alojada nuestra música, pero recuerden que podemos agregar secciones de video y fotografía también …



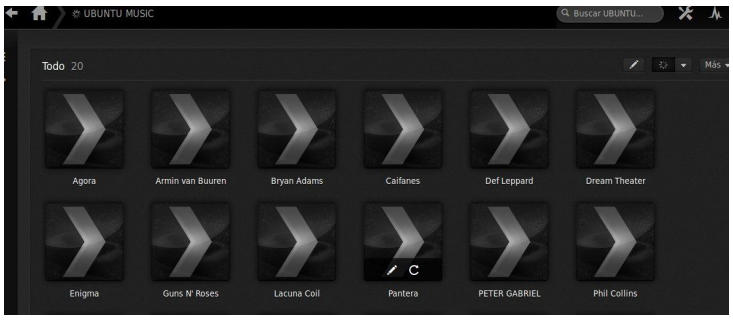
Aquí en esta ventana, como pueden ver, se está agregando la biblioteca musical que tengo almacenada en disco duro, el tiempo de carga dependerá de cuántos archivos de audio

seleccionemos ..

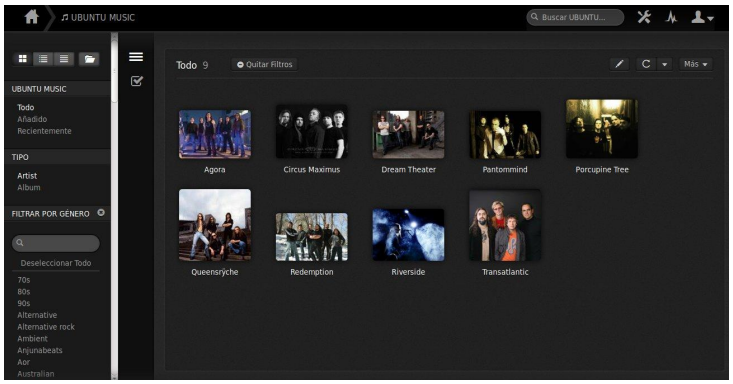




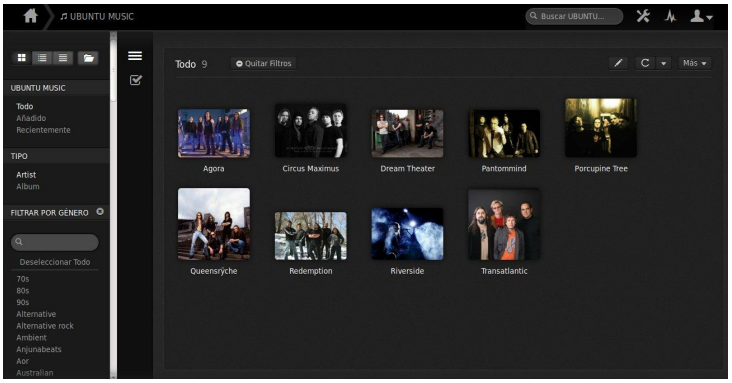
Una vez agregada toda nuestra biblioteca de audio chequen lo bien que lucen las carátulas de los discos, a un lado tenemos varias opciones para personalizar la vista de nuestro contenido y que podemos organizar por género, artista, álbum, etc.



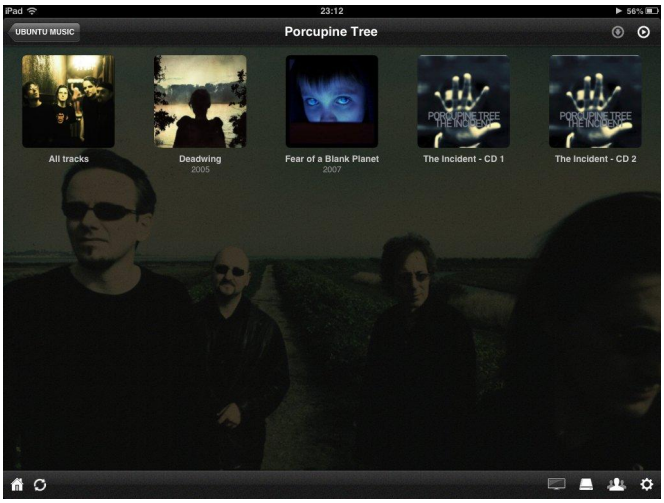
También tenemos una breve información de la banda que se está reproduciendo, esto me recuerda al grandioso y hoy muerto Songbird.

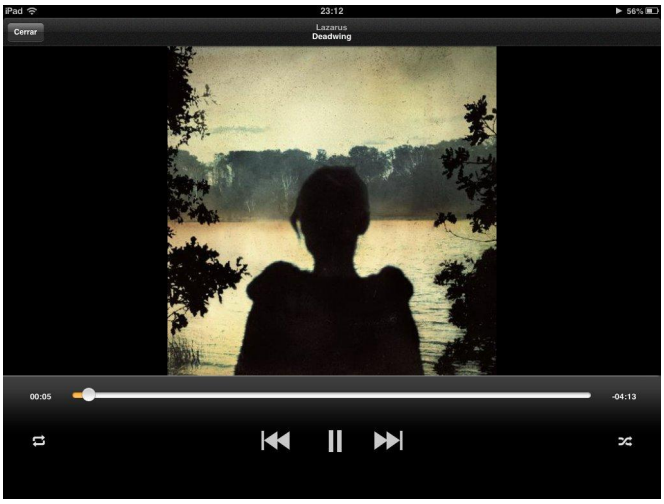


Ya tenemos nuestro “ server “ de audio montado y ahora procedemos a la descarga del cliente, en este caso lo voy a hacer en una tableta, por supuesto, como lo mencioné antes, lo pueden hacer en un smartphone.



Ya me bajé el cliente y una vez que introducimos nuestra cuenta, la misma que se creó previo a la descarga del ejecutable veremos en unos momentos cómo aparece todo el contenido de audio y listo, podemos reproducir con una fluidez muy buena .



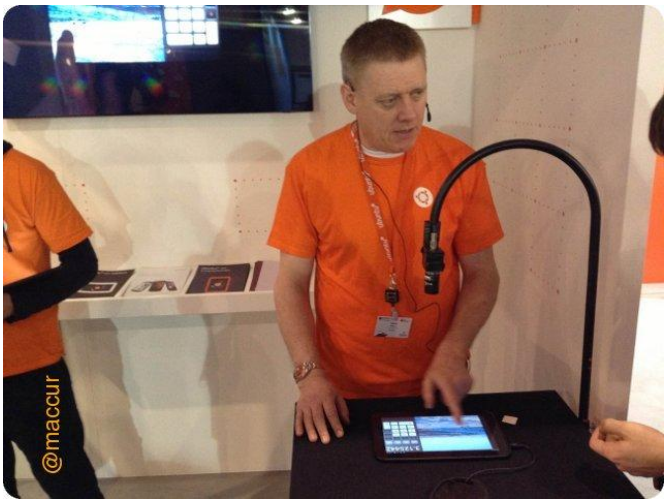


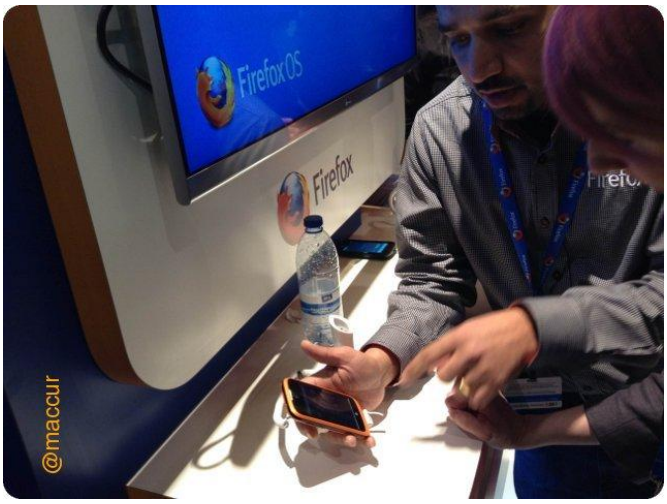
Me encanta como luce este programa en los dispositivos, ahora que si les alcanza para un gran disco duro que pueda conectarse a la red entonces tendrán todo una rockola para disfrutar, como les

comenté, este programa se puede explotar y sacarle todo el jugo posible … Nos leemos en la próxima señores

SISTEMAS ALTERNOS
Samuel Morales Cambrón
smorales@sistemasalternos.com.mx
@smcambron

Fotos del Mobile World Congress 2013 Tomadas por Mauro Accurso @maccur





TUXINFO

WWW.TUXINFO.COM.AR