

Tuxinfo

f /tuxinfo
t @tuxinfo

una revista libre, para un mundo libre.

nro.58



Puppy-es Night beta7

Un pequeñín españolizado.

Alias en GNU/Linux.

Opinión

Comparando un elefante con una aspirina

Guía de OpenOffice: Calc

Evento

Flisol Tic Tac Cucuta Tecnológica 2013

Proyecto Educativo Childsplay

Cuando - es + en 3d

FliSol 2013

el encuentro libre de cada año en San Juan

Pay-per-Install

como veta de negocio del cibercrimen mundial

Virtualizando con KVM

Guía Inicial - Parte II

VPScontrol

Administración web para servidores virtuales XEN

Probamos una notebook PositivoBGH con Ubuntu Linux

Redes para las masas

Parte III

Me

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

Nuevamente tenemos el agrado de publicar un nuevo número de TuxInfo. En este caso el número 58, el cual viene plagado de notas en relación a nuestro mundo libre. En esta edición van a encontrar muchos títulos que nos estaban requiriendo desde nuestros medios electrónicos. Pero como si eso fuera poco también hacemos entrega de dos reportes sobre lo que aconteció en el FLISOL 2013. Para ello tenemos específicamente una de nuestro país -Argentina- y la otra de Venezuela, países en donde de alguna manera se les está imponiendo cierta fuerza al uso del software libre y Linux.

Cabe aclarar que en nuestro país hay un proyecto que está tratando de reflotar la instalación de GNU/Linux en el plan Conectar Igualdad, el mismo se llama Huayra, y se encuentra llevado adelante por personas íntimamente ligadas al Gobierno Nacional.

Desde TuxInfo nos mantenemos cautos a la hora de hablar de dicho proyecto, ya que hemos visto ciertos tintes que no están muy de acuerdo hacia la política del software libre; pero tampoco lo desestimamos ya que, después de todo, el mismo impulsa el uso de Linux y software libre en el aula Argentina. Igualmente estamos esperando a octubre, fecha que dieron (se pospuso la de marzo del 2013) como culmine para la implementación en todas las netbooks del país.

Mientras tanto hemos asistido a dos charlas en donde específicamente se menciona el plan (no el de las Pcs, sino el de la distro Linux) y la verdad no hemos salido muy conformes. Una de ellas se brindó en la Universidad

UTN de Buenos Aires, en donde el disertante no diferenciaba temas esenciales como los conceptos que debía cumplir el software libre y los que no. En la misma charla y ante las preguntas insistentes de los asistentes, tuvimos que intervenir para poder aclarar el punto consultado. Aclarado el tema todo siguió con total normalidad. No quiero dejar pasar la oportunidad de felicitar a los organizadores del FLISOL en CABA (Ciudad Autónoma de Buenos Aires) por la prolijidad, orden y excelentes contenidos brindados a los asistentes el día 27 de abril.

Como para redondear el editorial, les cuento que el número está plagado de notas interesantes tales como: Puppy-es Night beta7 - “Un pequeñín españolizado”; Alias en GNU/Linux; Comparando un elefante con una aspirina; (opinión) Guía de OpenOffice -Calc; Evento: Flisol Tic Tac Cúcuta Tecnológica 2013; Proyecto Educativo Childsplay; Cuando – es + en 3D; FliSol 2013 el encuentro libre de cada año en San Juan; Pay-per-Install como veta de negocio del cibercrimen mundial; Virtualizando con KVM - Guía Inicial Parte II; VPScontrol - Administración web para servidores virtuales XEN; Probamos una notebook PositivoBGH con Ubuntu Linux; y Redes para las masas - Parte III.

Y como todos los meses, repetimos la misma convocatoria en donde podemos tener más sugerencias de ustedes y así adaptar los

Ariel M. Corgatelli
@arielmcoreg

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

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

Ariel M. Corgatelli



índice

2- Índice.

4- VPScontrol Administración web para servidores.

6- Redes para las masas Parte III.

14- PayperInstall como veta de negocio del cibercrimen mundial.

16- Entonces... ¿me repite la pregunta?.

18- Alias en GNU/Linux.

20- Cuando – es + en 3D.

25- Proyecto Educativo Chilpsplay.

28- Flisol 2013 el encuentro libre de cada año en San Juan.

30- Flisol en Tandil.

31- Flisol Tic Tac Cúcuta Tecnológica 2013.

32- Probamos una notebook PositivoBGH con Ubuntu Linux.

34- Puppies Night beta7 “Un pequeñín españolizado.

37- Comparando un elefante con una aspirina POR Claudio De Brasi.

38- Guía de OpenOffice - Calc.



VPScontrol

Administración web para servidores

POR Jorge Vaquero y Ricardo Marcelo Alvarez

Hoy en día la posibilidad de administrar un servidor GNU/Linux mediante una interfaz web ha pasado a ser una característica estándar. Un complemento básico que todo proveedor procura poner al alcance de los usuarios. Hasta hace algunos años se solicitaba el reboot de un servidor dedicado mediante un ticket de soporte y había que esperar con suerte unos cuantos minutos para solucionar un problema.

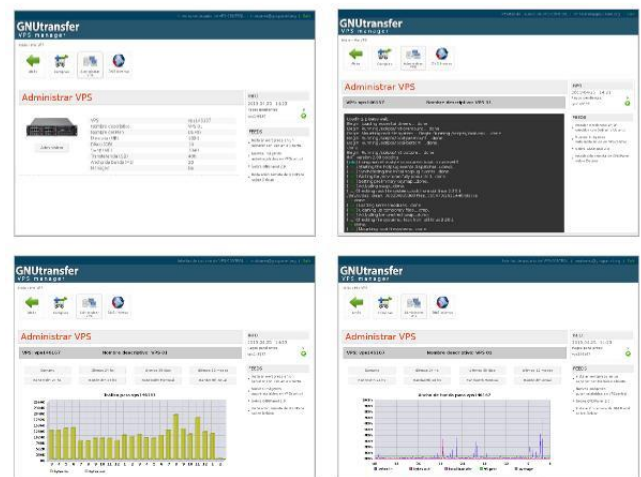
En principio administrar un servidor Linux sólo requiere de un acceso SSH a nivel root y una buena base de conocimientos, sin embargo herramientas de este tipo son sumamente útiles y permiten una mayor comodidad para realizar tareas administrativas y de mantenimiento.

VPSCONTROL es una de estas soluciones. Desarrollado en Argentina, es un nuevo panel de control para servidores virtuales XEN escrito por Ricardo Marcelo Alvarez (también creador de GNUPanel, uno de los pocos paneles de hosting libres disponibles para sistemas GNU/Linux). En este segundo desarrollo Ricardo ha volcado su mayor experiencia y muchos conceptos que formarán parte también de GNUPanel 2.0.



El programa consta de dos interfaces: ADMIN y USUARIO.

La interfaz **ADMIN** es usada por el proveedor de servicios. Desde esta interfaz se dan de alta los servidores físicos, se define la configuración de las máquinas virtuales, parámetros de red, medios de pago, todo lo estructural. Una vez activado el servicio esta interfaz permite al proveedor llevar un control de usuarios, pagos, recursos empleados así como atender solicitudes de soporte y seguir las estadísticas de consumo de las máquinas virtuales alojadas en cada nodo físico.



La interfaz **USUARIO** es naturalmente usada por los usuarios y cuenta con un amplio número de características que abarcan todos los requerimientos elementales de un administrador de sistemas:

- Botones Power ON / Power OFF / reboot
- Reinstalación automatizada del sistema operativo
- 6 distribuciones GNU/Linux a elección incorporadas y listas para usar: Debian, Ubuntu, CentOS, OpenSUSE, Fedora y Slackware
- Control de múltiples servidores desde la misma interfaz

- Esquema de particiones personalizado
- Soporte para sistemas ext3, ext4 y ReiserFS
- Estadísticas detalladas de tráfico y ancho de banda por servidor
- Gestión y control de pagos automatizado
- Envío y recepción de tickets de soporte
- Escalabilidad de recursos automatizada
- Soporte para paquetes de servicios adicionales (IP, bandwidth, backup FTP extra)
- Migración automatizada de máquinas virtuales entre servidores físicos
- Actualización automatizada del kernel
- Chequeo de discos
- Estructura de plugins para ampliar aún más sus funciones

VPSControl es una alternativa reciente dentro de este rubro de aplicaciones de control remoto para

servidores.

Su funcionamiento se apoya en una robusta base de datos PostgreSQL y está preparado para incorporar otras tecnologías de virtualización como OpenVZ o KVM e implementar la distribución de licencias de uso.

Enlaces de interés

- **VPScontrol Website**
- **Capturas de pantalla**
- **Tuxinfo Número 3, incluida nota de GNUPanel**

- <http://www.tuxinfo.com.ar/tuxinfo/?p=39>



CLA Linux Institute
Educación a distancia sin fronteras



CURSO

Debian Linux System Engineer

para APRENDER y CERTIFICAR

Estudia desde cualquier lugar del mundo, con el mejor material de estudio, actualizado y en español.

9 meses de duración * Inicia el 10 de junio

A cargo de **Fabián Ampalio** y el Ing. **Diego Córdoba**



Linux Professional Institute



UNIVERSIDAD TECNOLÓGICA NACIONAL

APRENDIZAJE
100%
GARANTIZADO



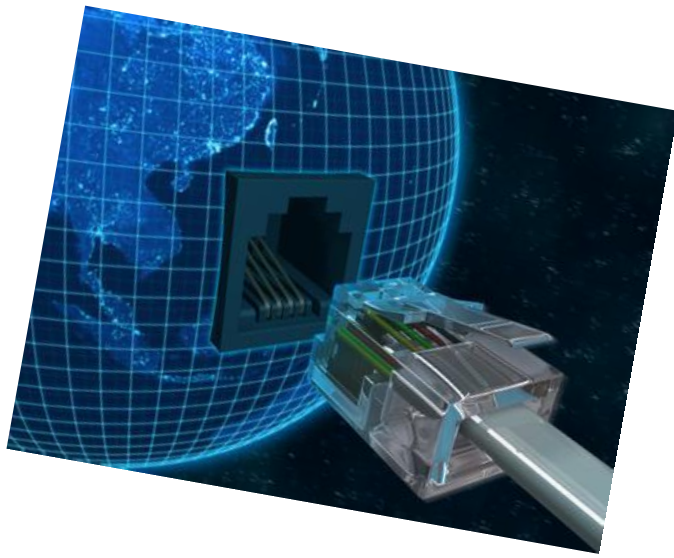


Informes y consultas: informes@carreralinux.com.ar | (+54.11) 4253.3362

Redes para las masas

- Parte III

POR Hernán "HeCSa" Saltiel



En los artículos pasados vimos elementos que nos permitieron conocer aspectos más que nada físicos de las redes de datos. Sus materiales, velocidades, y normas, entre otros.

En éste analizaremos algo que nos permitirá entender cómo es que dos o más máquinas logran comunicarse sin fallar en el intento. Veremos qué es el modelo de las siete capas OSI, y por qué es tan necesario que exista un estándar como ese.

Finalmente, y como para amenizar tanta teoría, nos pondremos a filtrar paquetes de red, entendiendo qué es lo que ocurre por debajo de una comunicación entre sistemas informáticos. ¡Manos a la obra!

Una arquitectura modelo

El conjunto de protocolos TCP/IP es conocido de esta forma por dos de sus más importantes componentes, que son el Protocolo de Control de Transmisión ("Transmission Control Protocol", o "TCP") y el Protocolo de Internet ("Internet Protocol", o "IP"). En algunos manuales viejos podremos encontrar para esto denominaciones tales como "Conjunto de Protocolos de Internet", ya que en los documentos oficiales, originalmente publicados, aparecieron con ese nombre.

El éxito de diseño de TCP/IP consistió en lograr la interconexión de redes de datos, referido en algunos documentos como "internetwork", o "internet", y entregar servicios de conexión a redes heterogéneas desde el punto de vista físico. Nótese que hemos escrito esta palabra

("internet") con una "i" minúscula, volveremos a esto más adelante.

Uno de los beneficios de ese diseño fue el de permitir que sistemas separados geográficamente puedan "conversar" entre sí, aún cuando esa separación sea inmensa. Hoy en día vemos cómo este éxito continúa de la mano de usar Internet (ahora con "I" mayúscula) y conectarnos a los sitios más remotos casi sin darnos cuenta.

Antes comenté que había que notar que estaba usando la palabra "internet" con minúscula. Pues bien, eso es porque esa palabra no es más que una contracción del término "interconnected network", y no hacen referencia a la red a la que nos conectamos para recibir nuestro correo electrónico, por ejemplo, llamada "Internet", con una "I" mayúscula.

Internet (con mayúscula) está compuesta por muchos elementos, como ser backbones, (redes grandísimas que fueron diseñadas en un principio para interconectar otras redes, por eso la traducción de backbone será la de "hueso de atrás" o "columna vertebral", que es el que sostiene a los demás huesos. En algunos manuales los veremos nombrados como "Puntos de Acceso a Redes", o "Network Access Points", abreviados como "NAPs", y en otros como "Puntos de Intercambio de Internet", o "Internet Exchange Points", en este caso abreviados como "IPXs"), redes regionales (éstas interconectan diferentes edificios, como ser empresas, escuelas, universidades, etc.), redes comerciales (aquí vemos redes que se suscriben a uno o más backbones para lograr su salida a otras redes) y redes locales (sólo interconectan las máquinas que se encuentran dentro de una organización).

Como vimos en las entregas anteriores, las redes generalmente estarán limitadas por la cantidad de personas que a ellas pertenecen (cuando decimos "personas", léase también "máquinas"), por la distancia geográfica

máxima que pueden cubrir, o por su grado de aplicación a diferentes ambientes. Entonces, podremos tener dos o más “internets” (con minúscula) que están interconectadas por un ruteador para formar una “internet” más grande. Será el mismo ruteador quien entienda la forma de retransmitir paquetes de red de una a otra parte de la red en forma transparente.

Un aspecto que brinda importancia a la forma en la que está diseñado TCP/IP es su capacidad de crear una abstracción estandarizada de los mecanismos de comunicaciones provistos por cada una de las redes, desde el punto de vista físico. Cada red tiene sus sistemas físicos de interconexión que le son propios, y que han sido adoptados en base a la tecnología disponible, el tipo de máquinas que deben interconectar, o la distancia que deben cubrir. Para cada una de esas interfaces físicas se programan funciones primitivas, específicas, pero que TCP/IP permite utilizar en su trabajo de interconexión de las porciones físicas de la comunicación para con las porciones lógicas.

Como la mayor parte de los programas de comunicaciones, TCP/IP está diseñado pensando en capas que funcionan casi como si se tratara de edificios con ascensores. Así es que tendremos un conjunto de capas ordenadas que regirán la forma en la cual las máquinas logran comunicarse, y que son muy útiles a los programadores para que no tengan que rearmar desde el punto inicial todo cada vez que deban armar un programa de comunicaciones. Divide y vencerás, es el lema, y en TCP/IP se pone de manifiesto de la mejor forma.

Imaginemos este escenario: vivimos en el séptimo piso de un edificio, y nuestro departamento tiene un ventanal que nos permite ver otros edificios. Un buen día vemos por ese ventanal una persona que por su grado de atracción no nos deja dormir en paz.

Para acercarnos e intentar la conquista tendríamos que abrir nuestra puerta, viajar en ascensor, salir por la puerta de nuestro edificio, caminar hasta el edificio de nuestro punto de atención, abrir la puerta de su edificio,

acercarnos a su ascensor, luego tocar el timbre, y si tenemos suerte, habremos recibido una respuesta a nuestra propuesta.

Igual funcionan las capas del modelo OSI, en el que nos adentraremos a continuación.

Haciéndose el OSI

OSI es la sigla correspondiente a “Open Systems Interconnections”, un producto que nació gracias a la entidad “International Organization for Standardization”, o “ISO”, aunque la sigla no siga estrictamente el orden de las palabras. Esta organización nació en 1926 como la “International Federation of the National Standardizing Associations”, o “ISA”, y cuyo principal objetivo era el de normalizar aspectos de ingeniería mecánica. Claro, en esa época, la electrónica no estaba tan difundida, y todo lo que luego vino estaba en un estado germinal absoluto.

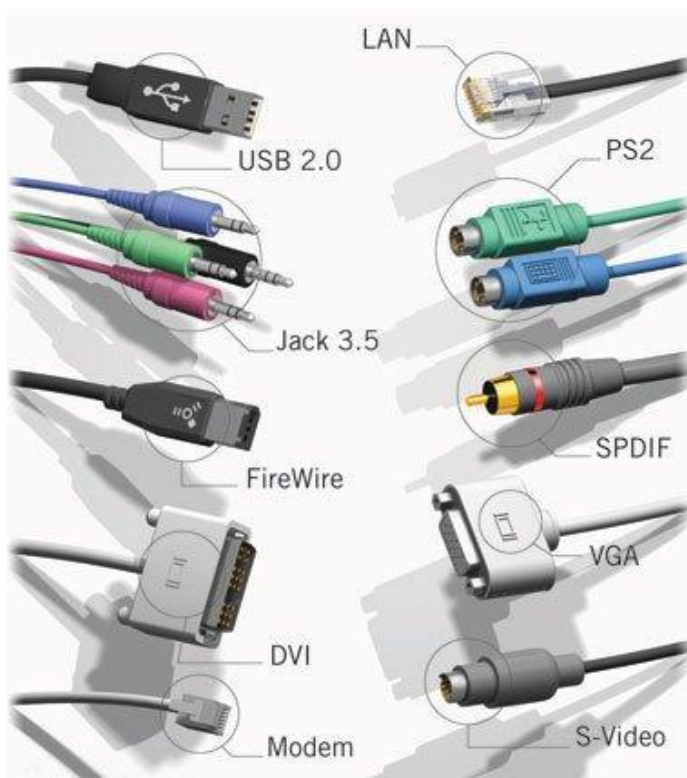
Como muchas organizaciones de ese estilo, cuando la segunda guerra mundial, en 1942, fue completamente desbaratada, para luego reorganizarse como la “ISO” en 1946, y recomenzar sus operaciones a principios de 1947.

Uno podría tener una idea bastante romántica de lo que es la “ISO” por pensar en su generación de estándares abiertos, pero lo cierto es que como muchas organizaciones, estuvo sostenida económicamente por empresas, muchas de las cuales estaban bastante deseosas de ver como estándar sus propios diseños, y así imponer en el mundo una u otra patente que les permitiera salir a flote. Hoy en día se establecen “estándares de facto”, y antes la tendencia era a tener un ente que los regule. Las caras han cambiado, las caretas son las mismas.

Pero dejando eso de lado, una de las buenas cosas que salió de la “ISO” fue el modelo “OSI” que divide la comunicación entre las siguientes capas:

- **Capa Física:** Capa 1. La capa física define las especificaciones físicas y eléctricas de los dispositivos de interconexión. Específicamente,

define la relación entre los dispositivos y sus medios de transmisión, como ser la fibra óptica, el cobre, y otros. Entre las cosas que podremos ver, en esta capa está la tensión (voltaje) que debe tener cada pata de un conector, cómo se definen las patas de esos conectores, su impedancia máxima, aspectos de tiempo de señales, y demás. Si no fuera por esta capa, oleríamos a quemado cada vez que conectáramos un nuevo dispositivo a nuestra máquina, ya que no es lo mismo conectar un cable de red que tenga unos pocos voltios, a un cable de la pared, con 220 voltios alternos, al menos donde yo vivo.



- **Capa de Enlace de Datos:** Capa 2. Provee los aspectos funcionales y procedurales para transferir datos entre redes de diferentes entidades, detectando y en lo posible eliminando cualquier tipo de error que se pudiera producir en el proceso de comunicación desde el punto de vista físico. Actualmente, sólo se maneja en esta capa el control de errores, y no el control de flujo eléctrico, como ocurría en un principio. Para entender mejor esta capa, imaginemos que enviamos un mensaje, y le agregamos datos de corroboración para que quien lo reciba pueda saber si lo que obtuvo es el mensaje correcto, o si tiene errores, y en este último caso, que cuente con los elementos para sobrellevar esta

corrección de errores en la mayor parte de los casos. Si enviara un mail, por ejemplo, en el que al final de cada palabra le agregara un número que representa la suma de los números de orden de cada carácter, podría pensar en un control de errores. “hola” sería seguido por 8 (posición de la “h” en el alfabeto) + 15 (ídem para la “o”) + 12 + 1 = 36, y el receptor recibiría “hola36”. Si le llega “holo36”, o “halo32”, sabría que algo anduvo mal en esa comunicación. Un punto importante a tener en cuenta es que esta capa sólo interconecta elementos de la misma red, no entiende cómo armar rutas entre redes. Elementos de red que se encuentran en esta capa son los hubs, o los switches de capa 2, ya que sólo tendremos control, en este caso, de elementos tales como la dirección “MAC”, o “Machine Address Code”.

- **Capa de Red:** Capa 3. La capa de red provee los elementos funcionales y procedurales para transferir secuencias de datos de longitud variable desde una fuente hacia un destino en diferentes redes, a diferencia de la capa anterior, que sólo lo hacía entre elementos de la misma red. Entonces, esta capa permite que funciones de ruteo tengan lugar, y por ende conocerá la forma de enviar paquetes de información desde una red a otra, así como recibir sus respuesta, y enviarlas a la máquina correspondiente. Los sistemas que operan en este nivel son los ruteadores (routers), que interconectan, por ejemplo, la red de nuestra casa con Internet. La capa de red, a la vez, se puede dividir en otras subcapas, que si bien son importantes, no aportarán mucho más a nuestro conocimiento de redes. Como se estarán imaginando, todo lo que se relacione con IP estará alojado en esta capa, ya que para poder armar rutas, debe haber un esquema de direcciones IP de las máquinas. Por supuesto, lo serán Ipv4, así como Ipv6, ARP, ICMP, IPsec, y otros tantos.

- **Capa de Transporte:** Capa 4. Esta capa permite la transferencia transparente de datos entre usuarios, entregando servicios de datos confiables a las capas superiores. Esta capa controla el grado de confiabilidad que un determinado enlace tendrá a través de controles de flujo, segmentación y desegmentación de paquetes, y control de errores. Veremos más adelante que algunos de los protocolos de comunicación que utilizaremos serán orientados a la conexión, pues entonces tendremos en esta capa la responsable de guardar un detalle del estado de cada uno de los segmentos recibidos, y de retransmitirlos en caso de errores. Por lo tanto, esta capa debe guardar también un detalle de cuáles son las

transmisiones de datos existentes, si fueron o no exitosas, y de transmitir los siguientes datos si los anteriores llegaron a destino. Como se podrán imaginar, todo lo que sea TCP será cercano a esta capa. También lo serán UDP, o SPX.

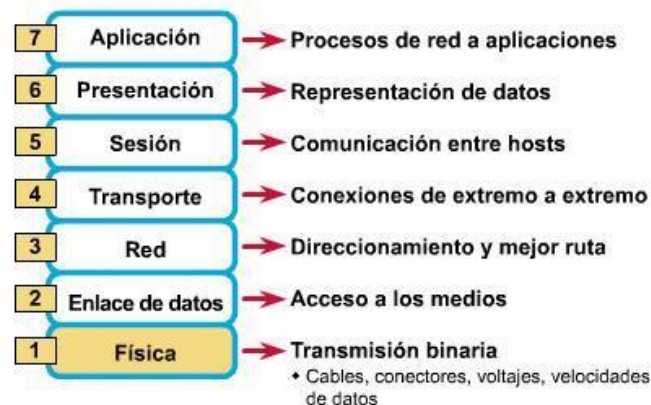
- **Capa de Sesión:** Capa 5. La capa de sesión controla los diálogos (si hacemos que las máquinas tomen consistencia humana, podremos hablar de este tipo de comunicación) o conexiones existentes entre distintas computadoras. Eso quiere decir que debe tener una tabla en algún lugar donde estén registradas todas las conexiones, y a la que deberá acudir cada vez que necesite conectar o desconectar alguna. Pero como es lógico, si esta tabla existe, tendrá también otros elementos, como ser detalles del tipo de comunicación establecida (si es half-duplex, full-duplex, etc.), metodologías de chequeo de conexiones, etc. Es muy común que esta capa esté implementada en aplicaciones que necesitan de la apertura o cierre de conexiones, como ser el caso de RPC, o “Remote Procedure Call”. Otros casos son NetBIOS, PPTP, o TLS/SSL.

- **Capa de Presentación:** Capa 6. Esta capa es un claro ejemplo de cómo se puede independizar al programador de aplicaciones orientadas a la red de tener que conocer todo sobre cada una de las anteriores capas, permitiéndole focalizarse sólo en lo que mejor (o peor) hace. La capa de presentación establece el contexto entre entidades a nivel de aplicación, entonces entendiendo su semántica, y mapeándolas para entregar datos en unidades encapsuladas siguiendo una norma estándar. Para que veamos esta capa con un poco más de claridad, veamos cómo una cadena puede ser transformada de un formato particular al ASCII, o cómo se pueden manejar las estructuras de datos existentes en archivos XML para ser generados e interpretados por las máquinas intervinientes en la comunicación. MIME o XDR son ejemplos de elementos que podremos encontrar en esta capa.

- **Capa de Aplicaciones:** Capa 7. Finalmente hemos llegado a la capa 7, la más cercana al usuario. Esta capa contiene al software de aplicaciones que luego hará uso de todas las demás capas inferiores para lograr una correcta y adecuada comunicación entre diferentes sistemas. Ejemplos de esta capa son el uso de HTTP, FTP, SMTP, IMAP, NNTP, NTP, NFS, o cualquier otro programa que permita la comunicación de cara al usuario.

A continuación, un diagrama que muestra todas estas capas:

Las 7 capas del modelo OSI



Es común que, como broma “nerd”, se hable de “errores de capa 8”. Como podremos imaginarnos, la capa 8 es el mismo usuario, con lo cual se hace referencia a que cuando algo falla, es el usuario quien debe ser reescrito, o puesto de acuerdo a algún estándar. Así nos divertimos los nerds, somos gente jocosa, qué se le va a hacer.



Por otro lado, si bien vemos una división estricta y bastante taxativa en lo que a cada capa refiere, veremos en la vida real que no lo es tanto. Inclusive, es muy común que a nivel de TCP/IP, algunas capas parezcan solaparse con otras, o que varias estén incorporadas en una misma, como pasa con las capas 4, 5 y 6, normalmente implementadas en nuestro stack como un único módulo, denominado “Capa de Transporte”, análoga a la que tenemos en el modelo OSI tradicional.



Paquete para su servidor

Ahora bien, ya sabiendo cómo funciona el modelo OSI, y cómo se mapea con el stack TCP/IP, podremos ver cómo se compone un paquete de red, y cómo se encuentran representadas, en él, cada una de las capas de este modelo.

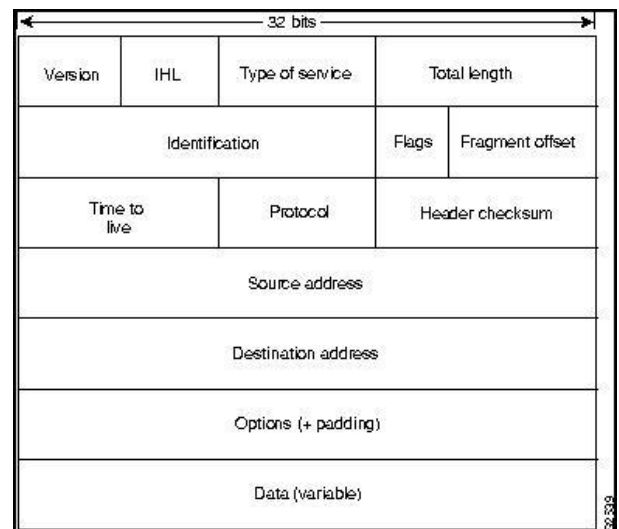
Un paquete de red es toda aquella unidad de datos transportada por una red de computadoras, sabiendo que los enlaces en sí mismos no transmiten paquetes, sino flujos de bits representados como “0’s” y “1’s”, y esos valores son conformados por impulsos eléctricos que dependiendo de si presentan o no tensión en sus conductores en un momento determinado así lo definirán.

Un paquete de red consiste de dos tipos de datos bien definidos, y que son los siguientes:

- **Información de control:** Provee la información que necesita la red para entregar los datos del usuario. Ejemplos son las direcciones de origen y destino, los datos de chequeo de errores y recuperación, información de secuencia de datos, etc. Esta información se encuentra en su cabecera y en su cola (conocidos también como “header” y “trailer”).
- **Cabecera:** La cabecera contiene instrucciones sobre los datos transmitidos por el paquete. Estas instrucciones pueden incluir la longitud del paquete (dado que algunas redes pueden transmitir paquetes de dimensión fija, mientras que otros se basan en la información de esta misma cabecera para establecer su tamaño), datos de sincronización,

número de paquete (cuando se reconstruya la información será esencial contar con esta información), información de origen y de destino del paquete.

- **Cola:** El trailer también es conocido como el pie del paquete, y normalmente contiene algunos bits que le explican al sistema que recibe la comunicación que ha llegado el final del paquete, así como alguna información de control de errores, como ser un CRC (“Cyclic Redundancy Check”, o chequeo de redundancia cíclica). Si los datos de control de errores no validaran al contenido del paquete, el sistema le pediría al que lo originó que lo reenvíe. Veamos una cabecera típica:



Notemos algunos datos interesantes: La dirección de origen utiliza 32 bits, así como la de destino. Recordemos que una dirección IP del protocolo Ipv4 se puede descomponer en cuatro octetos de bits, entonces nos cierra perfectamente la dimensión.

- **Datos de usuario:** Esta parte también se conoce como el cuerpo (“body”) o sencillamente como la sección de datos (“payload”). Es ésta la parte que contiene los datos que realmente se deben entregar al destinatario. Si el paquete fuera de dimensión fija, la parte del cuerpo que no contiene datos sería rellenada con ceros hasta cumplir con su tamaño.

Como ejemplo, consideremos un paquete de un mail (un mail de seguro implicará muchísimos más paquetes, pero veamos uno). Encontraremos en él una cabecera de unos 96

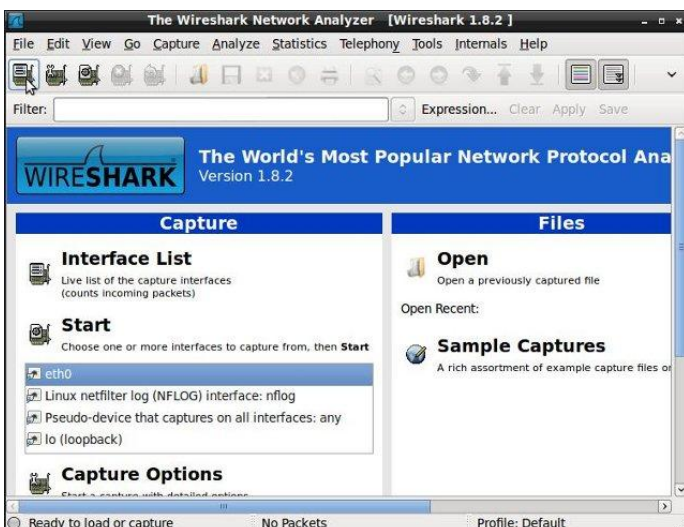
bits (32 bits de dirección origen, 32 bits de dirección destino, 8 para el protocolo, 16 para el número de fragmento, y 8 para otros elementos, como ser el TTL), luego un payload de 896 bits, y finalmente un trailer de 32 bits:

Header	Sender's IP address Receiver's IP address Protocol Packet number	96 bits
Payload	Data	896 bits
Trailer	Data to show end of packet Error correction	32 bits

Basta de teoría, pongamos las manos en la grasa

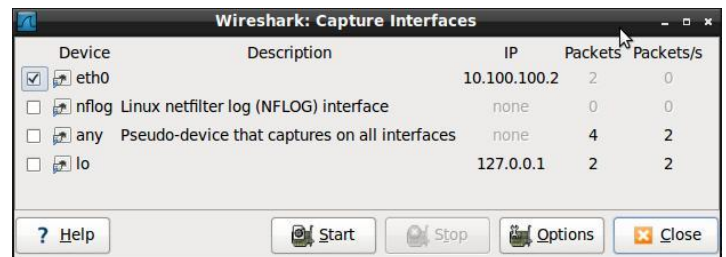
Ahora bien, imaginemos lo que pasa cuando ejecuto algo tan sencillo como ser un “ping”. En mi caso, ejecutaré, desde la máquina con dirección IP 10.100.100.2, un “ping 10.100.100.1”, mientras que ejecuto el programa “Wireshark” utilizando “sudo wireshark”, ya que necesitaré de acceso de root a las interfaces de red.

A continuación, la pantalla inicial de Wireshark al momento de ser abierto, cuando presionaré el botón que me permitirá seleccionar una interfaz de red, y comenzar a capturar paquetes:



NOTA: El programa Wireshark es un analizador de protocolos de red formidable. No es la idea de este artículo en particular explicar cómo

funciona, si bien en próximas entregas se hará especial hincapié en él, ya que en caso de problemas será una de nuestras herramientas más espectaculares. Se puede instalar en nuestras máquinas con “apt-get install wireshark”, “yum install wireshark”, o “pacman -S wireshark-gtk”, dependiendo de si la versión de sistema operativo es basada en Debian, en Fedora, o en Arch GNU/Linux.

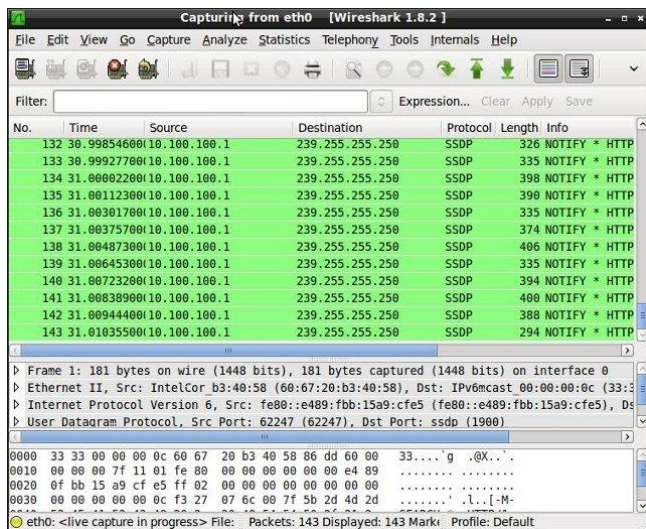


Entonces, seleccionaré mi interfaz eth0, y la pondré a capturar paquetes:

Ejecutaremos, en una terminal, “ping 10.100.100.1”:

```
hecsa@dshecsa01:~$ ping 10.100.100.1
PING 10.100.100.1 (10.100.100.1) 56(84)
bytes of data.
64 bytes from 10.100.100.1: icmp_req=1
ttl=64 time=1.02 ms
64 bytes from 10.100.100.1: icmp_req=2
ttl=64 time=0.732 ms
64 bytes from 10.100.100.1: icmp_req=3
ttl=64 time=0.736 ms
64 bytes from 10.100.100.1: icmp_req=4
ttl=64 time=0.705 ms
64 bytes from 10.100.100.1: icmp_req=5
ttl=64 time=0.722 ms
^C
--- 10.100.100.1 ping statistics ---
5 packets transmitted, 5 received, 0%
packet loss, time 4001ms
rtt min/avg/max/mdev =
0.705/0.784/1.027/0.124 ms
```

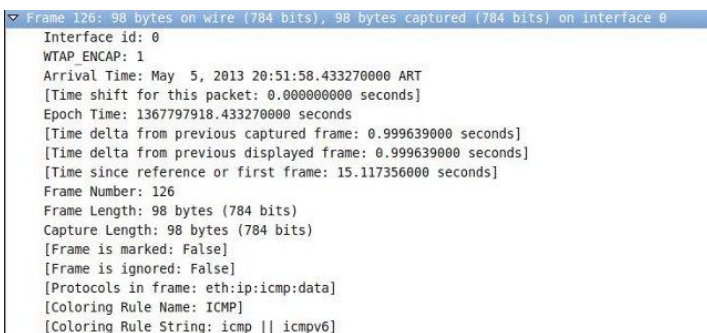
Luego de unos segundos de ejecución del ping, presionaremos “Control+C” para finalizar este programa, y veremos en la ventana de nuestro Wireshark cómo ha quedado logueada muchísima información, no sólo de nuestro ping, sino de cualquier cosa que haya pasado por la interfaz eth0:



Entonces, seleccionaremos alguna línea que contenga la cadena “ICMP” y “10.100.100.1”, para ver qué es lo que ha ocurrido.

Veremos que en la sección intermedia aparecen varias subsecciones desplegadas que nos entregarán información de lo que ocurrió cuando lanzamos el mencionado ping, en cada una de las capas donde intervino:

- **Capa 1:** En esta capa nos mostrará lo que ha ocurrido a nivel eléctrico en el cable, y la interfaz de red en sí misma. Encontramos que se han enviado 98 bytes (784 bits), y los mismos han sido capturados en la interfaz “0”. El número de “frame” es el 126, lo que me serviría, en caso de problemas de red, para saber si ese mismo número llegó al sistema “10.100.100.1”, que es el destino de mi “ping”. No profundizaremos en lo que es y significa WTAP_ENCAP, u otros elementos de esta sección, ya que eso nos llevaría a meternos en el código de wtap.c, y no es la idea volveros más locos de lo que ya están.

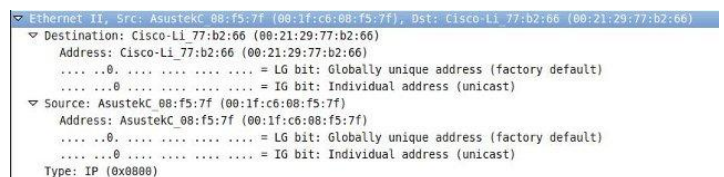


- **Capa 2:** Si revisamos lo que más arriba vimos, la capa 2 es la que ya comprenderá lo que es una dirección MAC, y si asociamos eso con lo que vimos en el artículo anterior,

sabremos que la dirección MAC tiene embestado un código que hace referencia al fabricante de cada una de las tarjetas de red intervinientes en la comunicación.

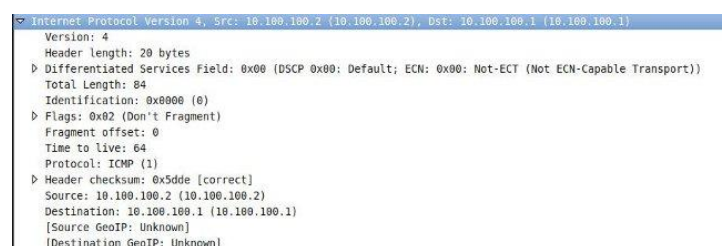
En nuestro caso, la máquina de donde sale el paquete de red tiene la dirección MAC 00:1f:c6:08:f5:7f, que se corresponde con una del rango asignado a Asustek, y por eso es que el programa Wireshark así lo reconoce.

Por otro lado, el primer punto que tocó el paquete de red fue un router Cisco Linksys, cuya dirección MAC es la 00:21:29:77:b2:66. Eso también lo podremos ver representado en esta parte de la pantalla.



Nótese que en esta sección no hay más elementos que los dos que se interconectan directamente. ¿Por qué? Si releemos la parte donde explicamos las características de la capa 2, veremos que aún no posee ningún elemento que nos permita ver todas las rutas utilizadas para llegar de nuestro origen a nuestro destino, que es, en sí, el equipo Cisco Linksys.

- **Capa 3:** En esta capa veremos a nuestro querido protocolo IP abrir sus alas con todo el esplendor, cual pavo real atrayendo a su pareja. A diferencia del pavo, con Wireshark podremos ver sus intenciones y contenido verdaderos, encontrando elementos que nos muestran las direcciones IP de origen, de destino, el tipo de protocolo (Internet Protocol Version 4), y otras tantas. Ya en este caso, nos muestra que el protocolo es el ICMP, que es el que se corresponde al “ping”. Nótese también que se han ejecutado pruebas en la cabecera del paquete de red, y se ha llegado a la conclusión de ser correcta.



- **Elementos directamente relacionados con el “ping”:** Veremos, en la siguiente parte de la misma sección, que se describe todo lo que ha ocurrido a nivel de ICMP, que es el protocolo que se utiliza cuando ejecutamos “ping”.

```
Internet Control Message Protocol
Type: 8 (Echo (ping) request)
Code: 0
Checksum: 0x2a51 [correct]
Identifier (BE): 21450 (0x53ca)
Identifier (LE): 51795 (0xca53)
Sequence number (BE): 3 (0x0003)
Sequence number (LE): 768 (0x0300)
[Response In: 127]
Timestamp from icmp data: May 5, 2013 20:51:58.433251000 ART
[Timestamp from icmp data (relative): 0.000019000 seconds]
Data (48 bytes)
Data: 08090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f...
[Length: 48]
```

Conclusión

Ya tenemos los elementos necesarios para entender claramente cómo funciona una red de datos, así como contamos con algunas incipientes herramientas que nos permiten determinar qué está bien y qué está mal cuando algo falla.

En las siguientes entregas nos adentraremos en comprender un poco más de TCP/IP, motivo por el que tendremos que activar nuestras

máquinas virtuales para poder jugar un poco con los diferentes protocolos de red.

Tendremos la oportunidad de entender qué es lo que hacemos cuando configuramos direcciones IP en nuestras máquinas, tanto desde el punto de vista estático como dinámico, veremos qué es un router predeterminado, e infinitas cosas más que nos permitirán, cada día, conocer más de lo que hacemos cuando hacemos, y cómo analizar una red con conceptos bien sólidos.

¡Nos vemos!



Hernán “HeCSa” Saltiel

AOSUG Leader

CaFeLUG Member

Twitter: @hcsaltiel

hsaltiel@gmail.com

<http://www.facebook.com/hcsaltiel>

<http://www.aosug.com.ar>



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

Pay-per-Install como veta de negocio del cibercrimen mundial

POR: Jorge Mieres



En la pasada edición comentábamos el impacto del malware diseñado para el robo de información sensible sobre los dispositivos móviles del tipo Smartphone, particularmente los que poseen alguna versión de Android como sistema base.

Para complementar el ciclo delictivo que se esconde detrás de estas maniobras, lamentablemente cotidianas, explicaremos cómo funciona una de las vetas de negocios clandestinos más grandes a nivel global: el pago por instalación o Pay-per-Install (PPI) y el pago por clic o Pay-per-Click (PPC) por sus siglas en Inglés; y cómo funciona uno de los esquemas de botnets más difundidos para plataformas móviles.

¿Cómo rompen la seguridad de tu Android?

Si bien existen varias modalidades que permiten romper la seguridad de los dispositivos con Android, una de las estrategias con mayor tasa de éxito es la propagación de códigos maliciosos específicos. Cuyo éxito se potencia al someter el equipo al proceso de “rooting” o “Jailbreak”.

El rooting de estos dispositivos proporciona, básicamente, la posibilidad de acceder a funcionalidades que antes estaban bloqueadas para evitar la manipulación arbitraria de las configuraciones del equipo. Es una medida de seguridad importante y de lo cual hablaremos en otra oportunidad.

Entonces, cuando el equipo es sometido a este proceso queda “con todas las puertas abiertas”. Fundamentalmente, cuando el usuario no posee experiencia desde una perspectiva técnica,

potenciando así la probabilidad latente de convertirse en una víctima más del aparato ciberdelictivo.

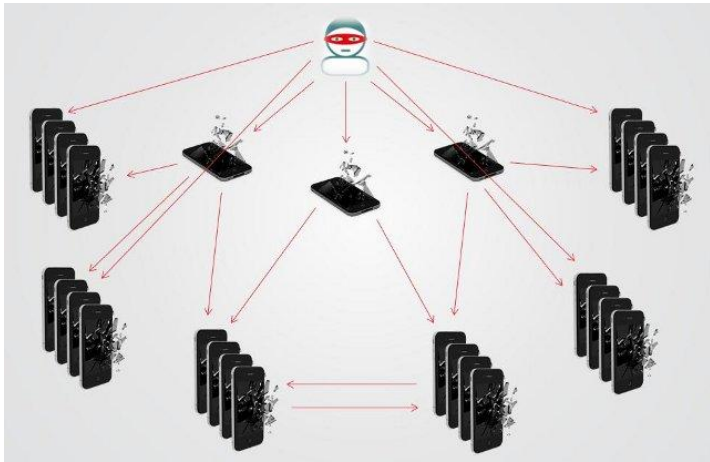
Bajo este escenario y siguiendo un orden cronológico, la consecuencia puede ser muy negativa ya que uno de los puntos críticos en este sentido es que, por ejemplo, el Smartphone se transforme en un “zombi móvil” no tan inteligente. Y a partir de ese momento el atacante podrá tener control total sobre el equipo.

¿¡“zombi móvil” no tan inteligente!?... ¡Sí! Básicamente formará parte de una botnet (conjunto de equipos infectados) conjuntamente con otros dispositivos de similares características que se encuentran en todo el mundo.

A diferencia de las botnets convencionales que reclutan equipos zombis desde sistemas operativos de la familia MS Windows o incluso basados en *NIX, las botnets basadas en smartphones poseen un condimento extra: “centinelas”.

Los equipos centinelas son puentes dinámicos y activos entre el botmaster (atacante) y los zombis (víctimas), a través de los cuales el atacante envía comandos a cada uno de los equipos que forman la botnet, para que realicen diferentes operaciones como por ejemplo, realizar una conexión clandestina contra otro C2 (centro de comando y control), descargar nuevas variantes del malware, robar información, monitorear el acceso a la banca en línea, etc. Sin embargo, la tarea principal de estos centinelas es la de evadir el rastreo del origen de la botnet por parte de los investigadores.

Habitualmente se trata de una botnet del tipo centralizada, pero como eso es excusa para otra edición por el momento nos conformaremos con visualizar de forma gráfica un simple esquema de la estructura de una botnet basada en dispositivos móviles, donde se observa que los tres equipos individuales son los centinelas.



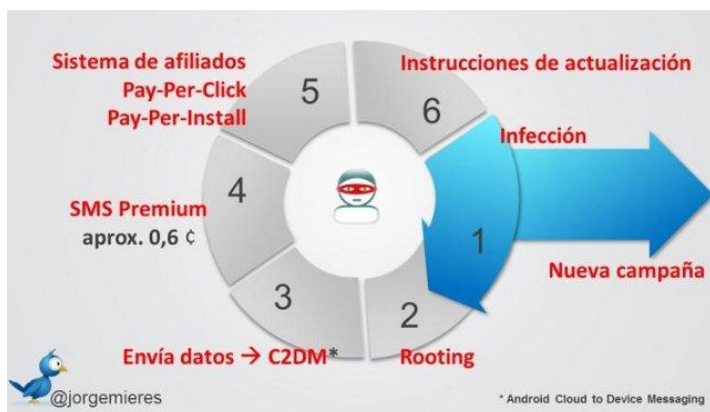
Por otro lado, la razón principal por la cual los botmaster dedicados intentan reclutar zombis móviles es generar diferentes vetas de negocio clandestino que alimentan el aparato productivo del ciber-crimen. Entre ellos, uno de los más difundidos es el PPI y PPC: Pay-per-install y Pay-per-Click respectivamente.

Sistemas de afiliados

Los sistemas de afiliados son esquemas de negocios muy difundidos a nivel global. Originalmente surgieron en Europa del Este pero con el tiempo se fueron expandiendo. Estos sistemas de afiliados proporcionan a los usuarios un conjunto de códigos maliciosos para su distribución. A cambio, cada usuario recibe un porcentaje de dinero por cada instalación de ese programa malicioso o por cada clic que se realice sobre los enlaces, también proporcionados por el sistema de afiliados, que direccionan el tráfico web del usuario víctima hacia la descarga de un malware.

Además, el sistema de afiliado o el atacante que se encuentra detrás de la maniobra, diseña el malware para que una vez que logre infectar un equipo, éste comience, entre otras cosas, a enviar mensajes de texto SMS del tipo Premium a números extranjeros. Aquí también se canaliza otra veta del negocio clandestino.

La siguiente imagen muestra el ciclo básico que recorre este tipo de negocios fraudulentos:



1. A través de una campaña de propagación, con un fuerte contenido de ingeniería social visual, el atacante disemina el malware o el enlace proporcionado por el sistema de afiliados. Se produce la infección.
2. Una vez infectado el equipo, el malware puede establecer una comunicación clandestina para descargar otras piezas de malware que se encargarán de realizar un “rooting” del equipo. Esta tarea también puede estar implícita entre las instrucciones del malware original.
3. Utiliza el servicio Android Cloud to Device Messagin (C2DM) para registrar bases de datos y habilitar la posibilidad de enviar mensajes de texto.
4. Envía de forma masiva mensajes SMS a números Premium. Por cada mensaje de texto enviado, el atacante (llámese botmaster) recibe otro porcentaje de dinero, que ronda los 0.6 centavos de dólar.
5. Se registra cada infección en la base de datos del sistema de afiliados y se asigna el porcentaje de dinero que corresponda al afiliado que ha conseguido con éxito instalar el programa malicioso o lograr que el usuario víctima haga clic sobre el enlace malicioso (según lo que corresponda).
6. El sistema de afiliado actualiza periódicamente el malware y los enlaces proporcionados para comenzar un nuevo ciclo de propagación. Este esquema representa solamente una de las tantas oportunidades de negocios que se gestionan desde la clandestinidad y a través de recursos underground que se encargan de alimentar el aparato delictivo que diariamente intenta golpear contra nuestra seguridad y, particularmente, nuestra información.

Jorge Mieres, Analista de Malware del Equipo Global de Análisis e Investigación de Kaspersky Lab.

Microsoft MVP Enterprise Security y fundador de MalwareIntelligence



Entonces... ¿me repite la pregunta?

POR Ma. Eugenia Núñez



Quizás una de las cosas que creo me diferencia como maestra de las que tuve en mi infancia, es la posibilidad de dar explicaciones a mis alumnos más allá de lo imaginable. Los docentes de antaño se sentían perturbados ante una posible pregunta que no tuviera respuesta. La no respuesta implicaba ponerse a la par del alumno a quien debía enseñarle el camino, mostrarle lo que no estaba visible. Como alumna, era frustrante no poder preguntar porque no había respuestas para mí del otro lado. Recuerdo una maestra que nunca logró entender mi dificultad para memorizar las tablas de multiplicar y a quien debía engañar aprendiéndolas para luego, simplemente, olvidarlas.

Por algún extraño motivo, nunca tuve temor a la no-respuesta porque soy una convencida de que el rol del maestro hoy en día no es enseñar a sumar, restar, leer o meramente escribir. Hoy un maestro es quien busca y ayuda a encontrar esas respuestas transformándolas en una construcción compartida. Y cuando se construye es imposible no involucrarse desde la más profunda subjetividad. Un maestro enseña sólo aquellas cosas que conoce y se encuentran dentro de su entorno subjetivo, cosas de las que está plenamente convencido y que comparte con pasión.

Trabajo como facilitadora pedagógico digital y como maestra de 3° grado en escuelas dependientes del Gobierno de la Ciudad de Buenos Aires y, desde ambos roles, me encuentro en permanente relación con los llamados "nativos digitales". En términos de **nativos e inmigrantes digitales**, mis alumnos serían nativos, nacieron y crecieron con la tecnología a su alrededor, analizan la realidad en forma no lineal y pueden hacer varias cosas al mismo tiempo. Sin embargo, tienen un problema: una maestra que es inmigrante digital, que trata de entender el mundo en el que se mueven, mostrándoles todo lo que no ven y que, de uno u otro modo, se están perdiendo. A

esto se le suma un agravante, es una mujer apasionada por el Software y la Cultura libre.

Para muestra solo hace falta un botón

Al comenzar el año, mis peques insistían en preguntar cuando empezarían a usar las netbooks. En períodos de conocer al otro, lo que menos necesitamos es la mediación de una máquina y fue por eso que llegamos invictos al mes de abril. Ya habiendo medido las distancias mutuas correspondientes, trajeron sus nets al aula. Desde el fondo, uno me preguntó: -Sé, ¿vamos a usar Ubuntu? Lo miré desconcertada y le dije: - ¿Ubuntu? Y me respondió: - Sí eso otro que tienen las nets. En ese momento caí en la cuenta de que quien hablaba era el hermano de un exalumno mío. En diciembre de 2011 había puesto a punto el Debian que traen las máquinas entregadas por el Gobierno de la Ciudad (dicho sea de paso, ese Debian merecería una nota entera), y lo había acompañado ese día poniendo en orden también su máquina. O sea, llevaba un año y medio usando Debian solito. Al instante saltaron otros tres peques contando que usaban distintas distros GNU/Linux en sus casas. La suerte estaba echada, ellos marcarían el camino.

Si hay algo que admiro de los "nativos digitales" es la capacidad que tienen de pasar de un sistema operativo a otro pensando sólo en las ventajas que le ofrece la posibilidad de elegir. Si un programa es fácilmente configurable en Debian, ¿por qué usarlo en Windows donde no encuentro mi trabajo después? Si puedo navegar con rapidez, ¿por qué voy a tolerar no poder hacerlo?

La idea del "No quiero esperar, lo quiero ya porque si espero se me va", quizás sea algo incomprensible para personas de más de 30 años. Esa cosa que tiene la inmediatez... que nos parece tan insensata...

Pero en el fondo tienen razón. En un mundo donde todo ocurre tan vertiginosamente, la única manera de desacelerar es cambiando con la misma rapidez, regulando el tiempo según las propias necesidades y eso es, ni más ni menos, lo que los chicos encuentran al usar un sistema operativo libre. No hay ventanas preguntando cosas incomprensibles cuando instalo algo, el tiempo es el que demoro en hacer un clic.

Muy distinta es la actitud de un adulto sin deseo. Abre un navegador, lo mira, espera, se toma en par de mates, que sigue esperando hasta que por fin algo aparece "un algo" en la pantalla. Cuando los miro de lejos no entiendo como pueden estar dispuestos a malgastar de ese modo su tiempo en esperas que solo conducen a tiempo malgastado. Lo peor de todo es que ante la mínima posibilidad de cambio se les eriza la piel y su NO es un NO rotundo. Pero la justificación es más terrible todavía y es frecuente escuchar frases tales como: "A mi no me cambies nada porque sino no lo entiendo" o "no me vengas con esas cosas raras que después no encuentro nada".

Después de muchos años de descreer del concepto de nativo digital asumo que existen aunque no son lo que muchos creen. No lo saben todo pero tienen deseos de aprender. No les importa si es fácil o difícil, si es mejor lo toman. Son críticos y sinceros, no esperan la perfección mientras les permita llegar a su objetivo. Conviven con gadgets y están conectados al mundo con un clic pero desconocen lo que hay del otro lado.

Y en medio de todo esto aparezco yo, una maestra inmigrante digital tardía que lleva sólo 12 años al lado de una computadora, la mitad de ellos como

hacktivista del Software Libre. ¿Bicho raro como pocos? Sí, así es. En definitiva, ser este bicho raro, marca la abismal distancia entre la maestra que soy y aquellas que otrora tuve que padecer. Para mí las preguntas son puertas que abren más preguntas, los errores pasillos que conducen a alguna respuesta y el conocimiento una construcción colectiva que crece día a día junto con mis peques.

Entonces... ¿me repite la pregunta?

Ma. Eugenia Núñez

@euge_nunez

<http://demasiadoalup.com.ar/>



"Digital Natives, Digital Immigrants" de Marc Prensky en "On the Horizon", MCB University Press, Vol. 9 No. 5, October 2001

Zimbra™
Collaboration Suite
Linware
www.linware.com.ar
zimbra@linware.com.ar

En cualquier lugar, en cualquier máquina

Somos una empresa líder en soluciones OpenSource y contamos con más de 5 años de experiencia instalando servidores de colaboración Zimbra.

vmware®
Business Partner

zimbra@linware.com.ar
+54 (011) 60090219
+54 (351) 5891012
+56 (2) 5952714



Este mes decidí dejar de lado los tutoriales de instalación para mostrarles algo realmente sencillo y que muchos puede ser que no sepan. Me estoy refiriendo a los Alias en los comandos de las terminales. Muchas veces cuando queremos convencer a alguien de usar un sistema operativo abierto y libre (léase cualquier distribución GNU/Linux), los podemos llegar a convencer por el lado donde no hay que bajarse ningún crack, la mayoría de los programas son gratuitos y se actualizan constantemente y el SO es casi libre de virus, pero cuando llega el momento de explicarles cómo usar la terminal, todo cambia. Un simple comando como “sudo apt-get update” o “yum -y check-update”, puede tirarnos todo para atrás. Claro, se le podría indicar a la persona que use el gestor de actualizaciones que trae cada sistema, pero todos sabemos que en algún momento esa persona va a tener que usar la terminal.

Es por eso que para que dichas personas se sientan más a gusto con el sistema operativo que quieren usar y para simplificarles un poco la vida es que existen los Alias.

Básicamente, los alias son “comandos sencillos” donde se enmascaran “comandos complejos” (por si lo queremos llamar de alguna manera). Podemos hacer que un comando como “sudo apt-get install <aplicación>” puede cambiarse a “instalar <aplicación>”.

Para hacer esto, vamos a simplemente editar un archivo que tenemos en nuestro sistema llamado .bashrc . Lo podemos hacer desde una terminal usando nuestro editor preferido.

Nosotros vamos a editarlo usando nano:

```
" $ nano .bashrc "
```

Alias en GNU/Linux

POR Natanael Andrés Garrido

En la misma terminal se abrirá el archivo el cual ya va a tener unas líneas de código escritas. Vamos al final del archivo y vamos a poder añadir los alias que queramos.

Acá les dejo unos ejemplos con los comandos habituales:

Ubuntu y derivados:

Actualizar el sistema:
alias actualizar='sudo apt-get update
&& sudo apt-get upgrade'

Instalar un programa/paquete :
alias instalar='sudo apt-get install'

Nota: Lo usamos así: instalar "paquete"

Desinstalar un programa/paquete
[desinstalar]:
alias desinstalar='sudo apt-get purge'

Nota: Lo usamos así: desinstalar "paquete"

Nautilus con permisos de superusuario
[nautilus+]:
alias nautilus+= 'sudo nautilus'

Nota: También podéis usar nautilus + directorio

ArchLinux y derivados:

Actualizar el sistema:
alias actualizar='sudo pacman -Syu'

Instalar un programa/paquete :
alias instalar='sudo pacman -S'

Nota: Lo usamos así: instalar "paquete"

Desinstalar un programa/paquete
[desinstalar]:
alias desinstalar='sudo pacman -Rsn'

Nota: Lo usamos así: desinstalar "paquete"


```
Terminal - neoranger@neoranger-Satellite-C645:~
Archivo Editar Ver Terminal Ir Ayuda
GNU nano 2.2.6 Archivo: .bashrc

/etc/bash_completion
fi

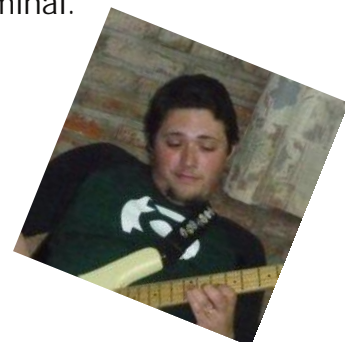
alias actualizar='sudo apt-get && sudo apt-get upgrade'
alias instalar='sudo apt-get install'

[ 110 líneas leídas ]
AG Ver ayuda  AO Guardar  AR Leer Fich  AV RePág.  AK Cortar Tex  AC Pos actual
AX Salir      AJ Justificar  AW Buscar   AV Pág. Sig.  AU PegarTxt  AT Ortografía
```

```
Terminal - neoranger@neoranger-Satellite-C645:~
Archivo Editar Ver Terminal Ir Ayuda
bash: alias: sudo apt-get install: no se encontró
neoranger@neoranger-Satellite-C645:~$ actualizar
[sudo] password for neoranger:
```

Espero que usen esta función que tenemos del sistema y que poco se conoce y que también sea una manera de que la gente nueva le pierda miedo a usar la terminal.

Hasta el próximo número.



Como vemos el comando se simplifica en gran parte. Lo bueno es que a todo comando en el que le pongamos "sudo" y usemos el alias, el sistema nos va a pedir la contraseña, así que por eso no nos preocupemos, la seguridad en el sistema con los alias no la vamos a perder. El alias puede tener el nombre que deseemos pero se sugiere que sea algo acorde a lo que es el comando para no confundirnos. En el caso de que el comando no funcione, lo más probable es que debamos reiniciar el sistema.

Natanael Andrés Garrido.
Twitter: @NatanaelGarrido
G+: Natanael Garrido
Web: www.neositelinux.com.ar

**CLA Linux Institute**
Educación a distancia sin fronteras

Curso CARRERA LINUX

Linux Principiante + Operador Linux + Administrador Linux + Redes Linux

- + Certificado de Asistencia
- + Certificado de Exámen CLA
- + Workshops LPIC 1

* 3 meses de duración *

* Inicia el 10 de junio *

\$990.- + I.V.A.
(Pesos Argentinos)

Promoción CLA & TUXINFO
Curso de Virtualización ¡SIN CARGO!

Linux Professional Institute



APRENDIZAJE 100% GARANTIZADO

Informes y consultas: informes@carreralinux.com.ar | (+54.11) 4253.3362

Cuando – es + en 3D

POR Pablo Emanuel "Timbis" Lopez

Es muy común pensar que en el mundo 3D siempre es mejor un personaje o un mundo prácticamente **real**. Ejemplos como Avatar dejan en claro esta afirmación, sin embargo no todos tienen una super-computadora capaz de procesar **miles de millones de vértices con efectos de simulación**.

Si recién empiezas en el mundo de Blender, este artículo te va a ayudar a no **“enfrascarte”** en que sólo vale la pena aprender a modelar objetos que parezcan reales y a texturizarlos de igual manera. O pasar horas y horas creando detalles y efectos visuales que lleven mucho tiempo de renderizado.

He visto cortos y películas en las cuales **el estilo de 3D lo es todo**, muchos de ellos muy simples. Estilos que nunca se nos hubieran pasado por la cabeza. **La idea es hacer volar la creatividad** en el estilo que queremos aplicar a nuestro 3D, y que al mismo tiempo sea **rápido de renderizar**.

Vamos a por los ejemplos:

Pocos vértices/caras

Los personajes y objetos tienen poco detalle.



<http://www.youtube.com/watch?v=A5-24n8y2uo>



<http://www.youtube.com/watch?v=yRTzEVAT8rI>

Entorno de pocos vértices/caras

Cuando el fondo no se lleva todo el detalle de modelado.



<http://www.youtube.com/watch?v=6PC0e7EsBxo>



<http://www.youtube.com/watch?v=UpmYzoPwIwA>

Toon

El texturizado no requiere alto detalle de luces, sombras, textura, etc.



<http://www.youtube.com/watch?v=neMxsBvFNCE>



http://www.youtube.com/watch?v=piHCwU6h4_o

Dibujo sobre 3D

Esta es una técnica que se va a poner de moda (seguramente), renderizar el 3D y dibujar los detalles a mano.



<http://www.youtube.com/watch?v=TZJLtujW6FY>

Cuando la textura lo es todo

Los personajes y objetos carecen de detalles, de esto se encarga la textura.



<http://www.youtube.com/watch?v=hXMdFGxrWik>

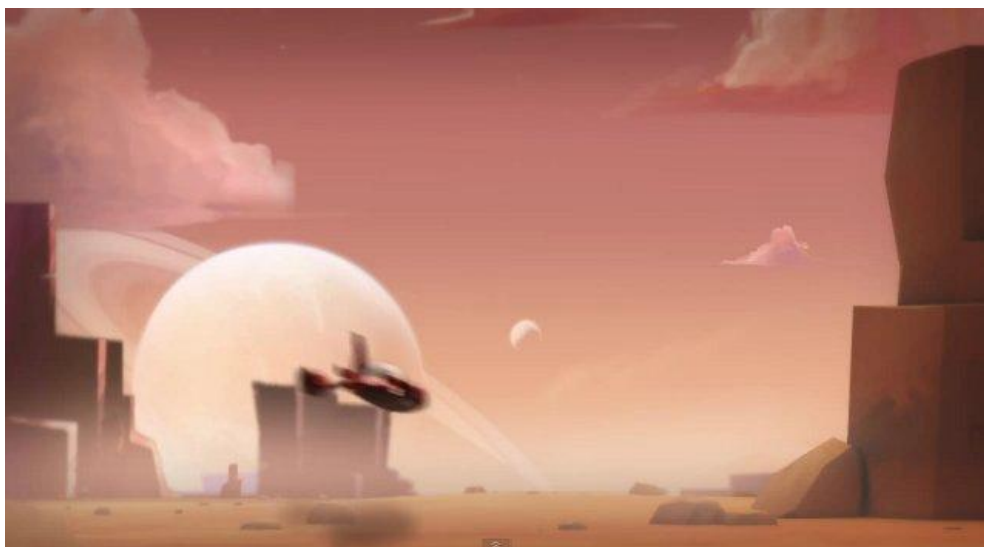
La textura es muy simple, pero sin embargo aporta mucho.



<http://www.youtube.com/watch?v=-yt7hUci82w>

Paisaje sobrio pero cuidado

Los fondos que más me gustan, son muy sobrios, pero llenan.



<http://www.youtube.com/watch?v=qdoQ-E4bBIM>

En esta categoría entrarían los cortos de Pablo Vazquez, genio total.



<http://www.youtube.com/watch?v=JOhiWY7XmoY>

Mezcla 2D/3D

Tal vez uno de los estilos más difíciles de que encajen. Requiere alguna experiencia en el dibujo.



<http://www.youtube.com/watch?v=EuSSC0Lv8oI>



<http://www.youtube.com/watch?v=SJWAeYYLBOE>

<http://www.youtube.com/watch?v=kRy05gCWQJ8>

Sobre Video

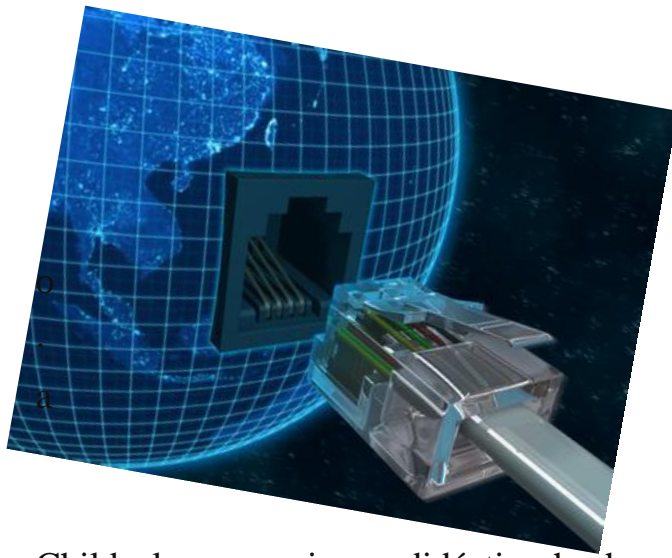
En este caso necesitarías un buen video para realizar camera mapping y object tracking, pero es una buena idea.

Seguramente existen muchos más estilos de 3D que permiten crear excelentes cortos/películas economizando recursos, como dije antes, este es un artículo que tiene como objetivo incentivar a la creatividad.

Me encanta ver cortos y películas 3D, por lo que si tienes alguno interesante puedes compartirlo en mi twitter.



Pablo Emanuel "Timbis" Lopez
@timbislopez



Childsplay es un juego didáctico hecho para niños de entre 2 y 7 años con el cual aprenderá a conocer el teclado, el ratón y sobre todo las letras, entre otras cosas, con una grandiosa selección de juegos didácticos que harán que tanto padres como hijos e incluso toda la familia se diviertan al máximo y de esta forma puedan dar sus primeros pasos con la informática.

Características principales de Childsplay

- Realizado en Software Libre.
- Soporte para varios idiomas.
- Registro de datos para monitorizar el progreso de los niños, a nivel local o en la red.
- Un conjunto de informes en formato de Open Office para imprimir.
- Buen soporte por los desarrolladores y traductores.
- Soportado para diferentes sistemas operativos: GNU/Linux, Windows, Mac.
- Ventana de trabajo orientada a objetos para desarrollar actividades fácilmente en Python/PyGame.

Los niños de entre 2 y 7 años de edad, irán aprendiendo a relacionar las letras que ven en el teclado, entre los 14 juegos que nos ofrece podemos encontrar:

- Memoria
- Memoria de letras minúsculas
- Memoria de letras mayúsculas
- Memoria de números
- Entrenamiento del ratón con acuario

Proyecto Educativo Chilpsplay

POR Nuady Villarroel Urquiola

- Asociación de sonidos y palabras
- Entrenamiento del teclado
- Rompecabezas
- Asociación de sonidos y cosas
- Tarjetas
- Tenis
- PacMan de letras
- Billar

Otra característica importante de resaltar es que desde el momento de inicio del juego el niño pequeño tendrá que ir aprendiendo letras, pues le pide un inicio de sesión que realmente "no existe", sólo es para guardar y que el niño la memorice, este juego cuenta con 6 niveles de dificultad para cada juego, con posibilidad de guardar. Su característica multiplataforma hacen que Childsplay se coloque en uno de los tres mejores juegos didácticos e interactivos.



- Una vez ingresado el nombre de usuario en la pantalla podremos ver el conjunto de juegos que nos ofrece Childsplay, en la parte inferior izquierda podemos ver el nombre del usuario, en la parte inferior derecha un botón en forma de 'X' el cual es para salir de la aplicación.

- El siguiente juego es un Rompe cabezas, la intención es armar una imagen que se muestra con los bloques que se encuentran en la parte izquierda, la forma de lograrlo es dando click en uno de los recuadros de la parte izquierda y arrastrarlos hasta un contenedor en la parte derecha, colocando la pieza en el lugar correcto dentro del contenedor.



- Todo juego incluye los botones para cambiar de nivel en el juego, o ver las estadísticas del progreso en cada juego. En la siguiente imagen con una flecha azul marcamos el botón para cambiar de nivel en el juego, el botón simula un dado y cada número del dado se refiere al nivel del juego, cada juego tiene hasta nivel 6. Con una flecha amarilla hacemos referencia al botón de estadísticas para ver el progreso en el juego. También en la parte inferior izquierda podremos ver el nombre del usuario que está jugando en ese momento.



- La imagen anterior es un juego de entrenamiento de mouse, consiste en darle click a los peces que van nadando, al darle click el pez se desaparecerá de la pantalla, según el nivel lógicamente la dificultad o velocidad será mayor. El objetivo de este juego es que el niño o niña pueda controlar y sincronizar el movimiento del mouse con su mano y el puntero en la pantalla.

- Para ver más acerca del entorno Childsplay hemos preparado un video explicativo.

Para descargar Childsplay en su versiones para Linux y Windows en los siguientes link:

- Descargar Childs Play V1.5.1 (Windows).

<http://www.scenebeta.com/system/files/private/ChildsplayV151WinMulti.rar>

- Descargar Childs Play V1.5.1(Linux).

<http://www.scenebeta.com/system/files/private/ChildsplayV151LinuxMulti.tar.gz>



Nuady Villarroel Urquiola

@naudy

FliSol 2013

El encuentro libre de cada año en San Juan

POR Lozano Juan Pablo



Como todos los años, el 3er Sábado de Abril se realizó el Festival de Instalación de Software Libre en toda Latino América. En esta oportunidad en San Juan, el evento estuvo organizado por la asociación civil Xonda encargada de difundir el Software Libre.

El evento comenzó a las 9hs con una linda apertura, y durante la jornada se llevaron a cabo varias charlas sobre diferentes temas para incentivar a los participantes a usar software libre en su vida informática.

Los temas tratados a lo largo de la jornada fueron:

- Introducción al Software Libre
- Primeros pasos en Ubuntu Linux
- Sistemas de Información Geográfica
- ¿Qué es y cómo trabaja Linux Mint?
- Suite Ofimática LibreOffice
- Debian en el ambiente Odontológico
- Conociendo Fedora GNU/Linux
- Proxy SQL Squid
- Instalaciones
- Conclusión y Cierre



La primera charla empezó con una introducción al software libre donde se dieron

a conocer las 4 libertades que brinda frente al software propietario, los beneficios y ventajas de usar este tipo de software, la casi ausencia de virus y de falta de necesidad de desfragmentación del disco duro en sistemas operativos GNU/Linux, entre otros temas.

Luego se introdujo en los primeros pasos en Ubuntu Linux como sistema operativo, se informó qué es una distribución GNU/Linux, qué objetivo persigue Ubuntu, a quienes está orientado el sistema, una demostración a lo largo del escritorio, el uso básico, etc.

Habiendo hecho un vistazo por lo más primordial, se dio a conocer cómo funciona el software libre en el ciencia Geográfica, una breve introducción, algunos conceptos, varios programas para usar en el entorno y cómo se desenvuelven.



Para mostrar la gran variedad de cosas que hay en este mundo del software libre, tocaron el tema Linux Mint sobre el entorno KDE dando a conocer otras opciones, programas y formas de trabajar. También se explicó cómo nace esta distribución y su filosofía.

Luego de un receso, continuaron las charlas,

esta vez sobre la Suite Ofimática LibreOffice con una comparación frente a la famosa suite de Microsoft (Microsoft Office), el poder que brinda sobre las libertades que posee, una breve historia de cómo se desarrollaron las cosas con OpenOffice y cómo se produjo el cambio a LibreOffice.



Para entender el potencial del software libre, se disertó sobre cómo se usa el sistema operativo Debian GNU/Linux en un ambiente odontológico (de trabajo) para explicar las estabildades que brinda. Demostrando con un recorrido sobre su funcionamiento con el escritorio GNOME 2 las aplicaciones que se usan para trabajar y que conllevan a no depender de sistemas operativos propietarios.

Se posibilitó conocer a Fedora, su vinculación con la empresa RedHat, su forma de trabajar, su filosofía y una pequeña demostración sobre cómo llevar a cabo la Instalación de este sistema en una PC.



También se tuvo a un experto en servidores proxy el cual explicó cómo se usaba una

distribución de Linux para controlar el tráfico de Internet en ambientes corporativos, las ventajas que proporcionaba el usar SQL Squid para gestionar las conexiones a internet de las computadoras de una empresa.

Y con mucho entusiasmo se realizaron a lo largo de las charlas las instalaciones de sistemas GNU/Linux en notebooks y netbooks de los invitados que asistieron al FliSol de este año.



Finalmente se hizo entrega de Discos FliSol 2013, el cual tenían un contenido único: DVD Bootable con 4 sistemas GNU/Linux (Ubuntu 12.04, Linux Mint 13 KDE, Puppy Linux y Gparted Live CD) con un atractivo menú selectivo. En ese DVD además se proporcionaba software libre para sistemas Windows, entre ellos: VLC, Clementine, Audacity, Gimp, LibreOffice, Chromium, Pidgin, 7Zip, Supertux, SupertuxKart, Gedit, etc. Y varios manuales y tutoriales para aprender a usar GNU/Linux, instalarlo y usarlo.

Si aún no se ha enterado sobre este evento, lo invitamos a asistir al próximo FliSol el siguiente año, recuerde que encontrará gran variedad de charlas, materiales, instalaciones, etc. Para informarse sobre dónde y cuándo se efectuará la próxima FliSol visite la página: www.flisol.net elija su país y luego su provincia.



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



El pasado viernes 26 y sábado 27 de abril se ha realizado el Festival Latinoamericano de Instalación de Software Libre en Tandil.

El mismo fue organizado por el Centro de Estudiantes de la Facultad de Ciencias Exactas de la UNICEN (CEFCE), MPE-UNICEN y Centro libre, avalado por la Facultad de Ciencias Exactas UNICEN. Se dividió en dos jornadas, la del viernes 26 en el Campus orientada principalmente a disertaciones/talleres y la del sábado 27 en el Centro Cultural Universitario donde en concordancia con la fecha oficial se procedió con la instalación de software libre en general.



Las actividades realizadas fueron las siguientes:

Viernes 26, Campus UNICEN (Tandil)

11:00 hs Charla introductoria al software libre a cargo de Hugo Curti.

12:00 hs Presentación de Huayra, la distro del plan conectar igualdad. Mostrando aplicaciones exclusivas pensadas para esa distro y sus posibilidades.

15:00 hs Charla sobre línea de comandos a cargo de Hugo Curti

16:00 hs Charla sobre Arduino a cargo de Oscar Goñi y José Marone. Mostraron algunos proyectos que están desarrollando.

18:00 hs Sistemas de monitoreo, Nagios, a cargo de Juan Ignacio Colella

Sábado 27 en el Centro Cultural Universitario, calle Irigoyen entre San Martín y Belgrano



10:00hs a 17:00hs Jornada de instalación

También se ofrecía grabar una copia en cd/dvd/pendrive/tarjeta de memoria en versión instalable/live de la distribución preferida.

15:00hs Charla sobre cooperativas de desarrollo de software a cargo de Leandro Monk, integrante fundador de GCOOP (Cooperativa de software) y FACTTiC

Si desea más información puede ver el siguiente video

http://www.youtube.com/watch?feature=player_embedded&v=sMI1HA9mOmc



En la jornada del viernes la concurrencia estuvo protagonizada por grupos de estudiantes de escuelas secundarias con orientación en informática y comercio. En cambio las actividades del sábado tuvo presencia de público en general, mayormente gente ligada al área de la informática.

Conclusión: Fue una fiesta bien organizada, amena, con buenos disertantes, muy recomendada para cualquiera que desee tener un primer acercamiento al software libre.



Oscar Reckziegel
Tandil

Cúcuta Abril 27

Biblioteca Pública Julio Pérez Ferrero

Tecnologías para la Inclusión Social

Software Libre

El pasado sábado 27 de abril, se realizó el Flisol Cúcuta 2013 en las instalaciones de la Biblioteca Pública “Julio Perez Ferrero”, ubicada en la Av. 1, Barrio la Playa en la Ciudad de Cúcuta - Colombia.

El Flisol 2013 es organizado por Cúcuta Tecnológica (<http://www.cucutatecnologica.com>), un grupo de jóvenes emprendedores en el campo de la tecnología, orientados en la inclusión digital, libre acceso a la información y disminución de la brecha digital, promoviendo la cultura del software y redes libres, la implementación de las TICS en los procesos educativos y productivos de la región, todo bajo un compromiso con nuestra gente y donde manejamos cuatro pilares: comunidad, conocimiento, emprendimiento e innovación.

Muchos se preguntarán, qué significa encuentro Tic Tac: “Tecnologías de la Información y la Comunicación – Tecnologías para el Aprendizaje y el Conocimiento”.

Estos fueron los temas: “Espacios Públicos para Todos ESPPATOD@S”, “Primer traductor de español a lenguajes de señas”, “Las oportunidades del blog en procesos de participación social”, “Software Lector de Pantallas para las personas con discapacidad visual”, “TATISPROJET – Iniciativa que busca desarrollar aplicaciones cerebro computador para mejorar la calidad de vida de las personas con parálisis cerebral”, “Android ADK-Accessory Development Kit”, “Introducción a FireFox OS”, “LibreOffice para todos”, “Instalación de Software Libre en la cabeza para el cambio social”, “Multimedia en Linux”.

También durante todo el día se realizaron rifas entre los asistentes para motivar más a los participantes.

Muy sinceramente les quiero dar un fuerte agradecimiento a todos los amigos y amigas organizadores y colaboradores del Flisol Cúcuta 2013, porque gracias a su gran trabajo en equipo, han logrado seguir avanzando en la difusión del Software Libre; y también darle las gracias a todos por la ayuda prestada, y por esta invitación.



Naudy Villarroel Urquiola
twitter: @naudyu

Probamos una notebook PositivoBGH con Ubuntu

POR Ariel M. Corgatelli



Como ustedes ya saben, siempre estamos a la pesca de probar equipos bajo nuestro querido y libre sistema operativo GNU/Linux. Está más que claro que todos los equipos lanzados al mercado funcionan completamente con el sistema operativo de Microsoft, es por eso que cuando gestionamos un préstamo, lo que hacemos es tratar de exprimirlo a fondo y ver hasta qué punto el mismo equipo es compatible con Linux.

En este caso tuvimos en nuestro laboratorio una notebook de la firma PositivoBGH, de producción nacional, a la cual le instalamos Linux. Todas las pruebas realizadas, como así también la posibilidad de ver en vivo su funcionamiento desde el siguiente enlace.

<http://www.youtube.com/watch?v=qUnYIViN9TI>



Dicho equipo fue lanzado en Buenos Aires en octubre de 2012. Donde la empresa lanzó al mercado su línea de Notebooks S-600. La serie incorpora un nuevo diseño y cubre todo el espectro de necesidades de los usuarios, con discos entre 320 y 750 GB, y entre 2 y 8 GB de memoria. Además, todos los equipos cuentan con pantalla de 14" de Tecnología LED, Webcam incorporada y

procesadores Intel® Segunda Generación.

Características técnicas del Modelo S-660

Notebook Positivo BGH Modelo S-660

Procesador: segunda generación Intel® Core® i5 2430M

Memoria: 6 GB DDR3

Disco: 640 GB SATA

Pantalla: 14" Tecnología LED

Óptico: DVD+/-RW (Dual Layer)

Puertos: 3USB

Salida: HDMI

La prueba con Linux fue realizada con una versión LTS 12.04 de 64Bits; la cual respondió de forma completa en todo su hardware. Además realizamos el testeo con Knoppix 7.0.5. En ambos casos fue ejecutado desde un pendrive, funcionando sin ningún tipo de sobresalto, y luego se procedió a instalar el sistema operativo en un espacio libre del disco rígido.

El mismo funcionó de forma correcta, y el grub realizó sin problemas el doble boot, haciendo que el equipo inicie en Linux y en Windows, con las simples opciones al inicio.

Lo bueno del equipo.

Creo que el primer punto a reconocer de esta laptop 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 esta notebook, 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, los materiales que componen al equipo son resistentes, cuenta con un amplio teclado y además la estética del mismo es del tipo sobrio.

Lo malo del equipo.

No hemos encontrado muchos puntos negativos, quizás el más importante es el peso y la poca autonomía del mismo. Por lo tanto el equipo no cumpliría una exigencia de tiempo de uso clásico de cualquier usuario.

Puntaje

8/10

Compatible con Linux

SÍ

Enlaces adicionales

Video revisión que le realizamos al equipo

<http://www.youtube.com/watch?v=qUnYIViN9TI>

Información oficial del producto desde la web de PositivoBGH

<http://www.positivobgh.com.ar/#/productos>

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. Su costo es muy competente, y su hardware es ideal para el uso intensivo por un usuario de Linux.



Ariel M. Corgatelli
@arielmcorg





Puppy-es Night beta7

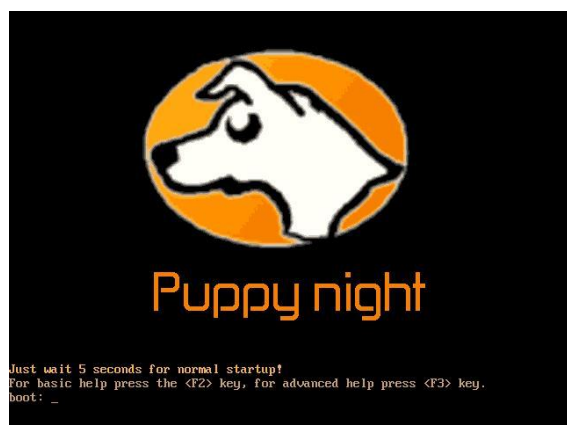
“Un pequeñín españolizado”

POR Juan Manuel Danza

En esta oportunidad mostraré a este pequeño sistema GNU/Linux, “puplet” (FORK) del tan conocido PUPPY Linux, es una distribución rolling release, para generar el núcleo desde el código fuente se usó T2 (es un conjunto de scripts de Bash utilizados para compilar una distribución GNU/Linux completa desde paquetes fuente: <http://www.t2-project.org/>); dicho proyecto comenzó en el 2012 desde el canal de IRC #puppy-es, donde un grupo de usuarios sintió la necesidad de tener una versión de Puppy Linux con soporte total y nativo de idioma español y soporte 3G directo.

En este proyecto nos encontramos con 2 ramas, la primera del proyecto denominada “Passaggio” (Puppy-es Passaggio 531-1.2RC, última versión) y la que trataré en este artículo, la nueva “Night beta7”. La cual tiene como novedad la utilización de un Kernel más moderno el 3.2.29 y basado en Puppy Linux 5.4.3 (Precise), a su vez basada en Ubuntu 12.04LTS, trayendo esto compatibilidad con muchos de sus paquetes.

Instalación



Pantalla de BOOT de night beta7

Lo primero fue descargar la imagen ISO de tan solo 190MB (http://mamalibre.no-ip.org/pub/puppylinux/puppyes_night_b7.iso) y paso seguido instalarla en un dispositivo USB de 2GB, que tenía abandonado, desde mi GNU/Linux con el programa UNetbootin

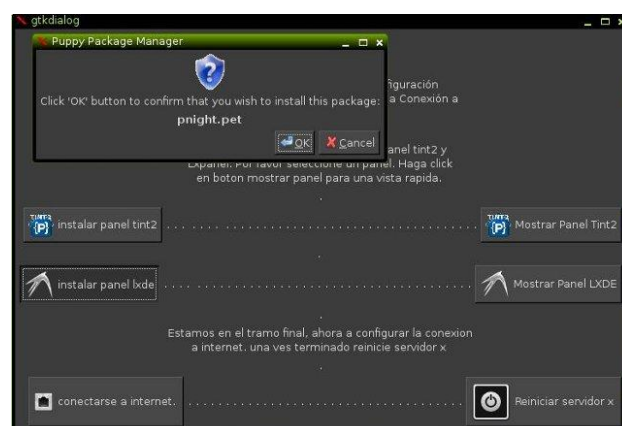
(<http://unetbootin.sourceforge.net/>), el mismo también disponible para Windows y MAC OS X.

Paso seguido se reinicia el ordenador con el dispositivo USB (configurar BIOS de la máquina para bootear desde dispositivos USB).

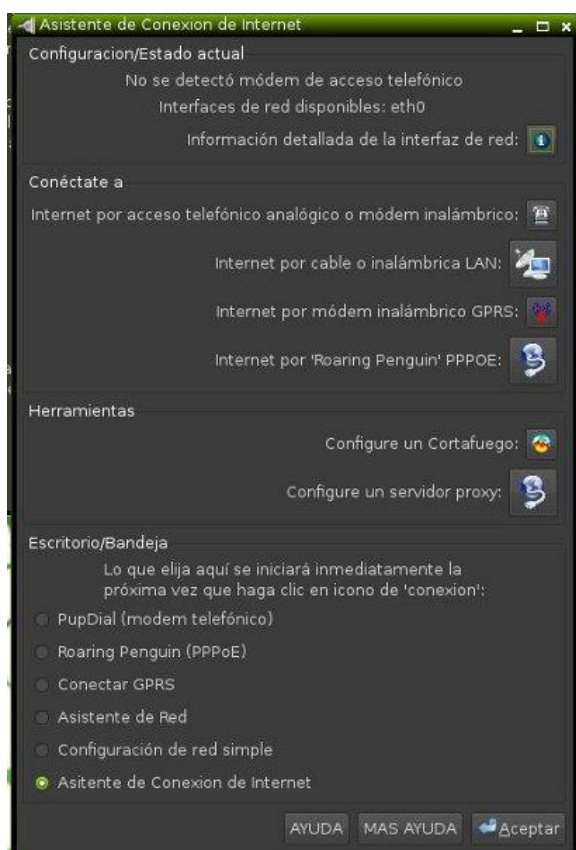
Nos encontraremos con la pantalla de boot, que si no tocamos nada o apretamos “enter” seguirá su camino hacia la apertura del sistema, el cual se encuentra sobre Openbox 3.5 lo que le da una velocidad increíble, con respecto a Puppy Linux se han mejorado y agregado librerías.

La primer pantalla de configuración que nos aparece es la de teclado, zona horaria, firewall, pantalla, hostname, etc, de una forma muy minimalista y rápida.

A continuación uno de los cambios más representativos, la elección de panel,



Asistente conexión internet



tint2 y el panel de lxde, en ambos casos se puede pedir una imagen de muestra para visualizar como quedan ambos paneles antes de la elección de alguno de estos; a mí particularmente me gusta tint2 por su minimalismo y su hermandad con Openbox. En la misma pantalla configuraremos la conexión de internet, lo cual nos lanza un asistente bastante cómodo, donde podremos configurar una conexión por cable o hasta por GPRS.

Terminadas las configuraciones básicas nos pedirá que reiniciemos el Servidor X, lo cual realizamos con un “click” y al volver veremos nuestra configuración con los paneles elegidos con anterioridad y con una nueva pantalla de configuración que destaca a esta “distro” y es la de poder elegir el administrador de archivos a utilizar, las propuestas son PCmanfm (el de LXDE) y ROX (por omisión y el más minimalista de los dos), como con los paneles también en esta oportunidad tenemos a mano dos imágenes que nos muestran como se ve cada uno.

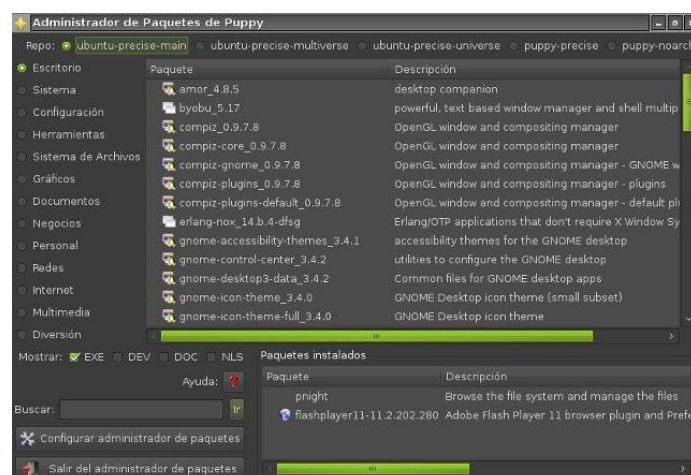
Ahora ya lo tenemos listo para usar, pero para que se guarden todos estos cambios,

procedemos a cerrar el sistema donde se nos preguntará si queremos guardar la sesión, esto vendría a ser un archivo de persistencia que se crea y se guarda, en este caso en el dispositivo USB (pero puede ser guardado en varios lados como en una partición de disco); tenemos también la opción de encriptar este archivo y de darle el tamaño que deseemos o directamente todo el espacio que nos quede libre y elegir la partición del mismo.

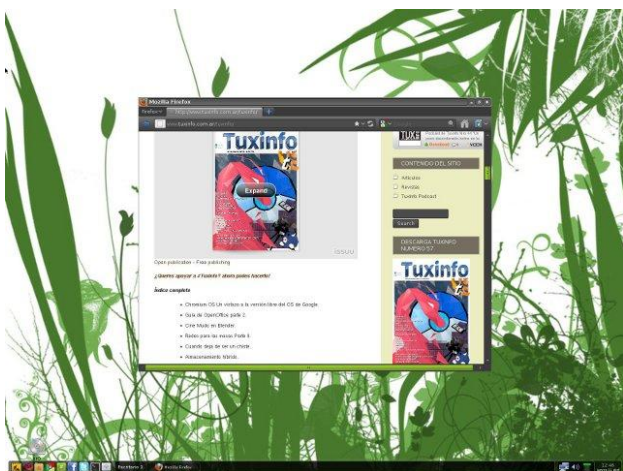
Esto provocará que en el próximo reinicio tengamos las mismas configuraciones como cualquier sistema convencional, y en el caso de encontrarse encriptado les pedirá el password previamente cargado para iniciar el sistema; recomiendo proteger el dispositivo aunque sea con una encriptación liviana que funciona bastante bien.

Dentro del sistema, podemos encontrar por ejemplo desde el menú de este, la descarga rápida de Adobe Flash Player, el navegador por omisión Firefox 13 (se actualiza automáticamente al ingresar en “acerca de”), configuración de apagado programado, temas de tint2, Abiword y Gnumeric, actualización del sistema, etc.

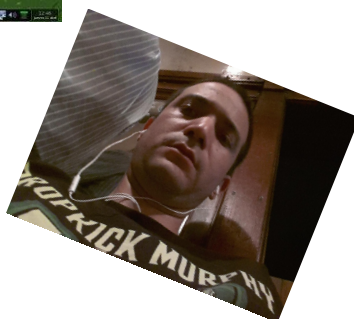
Con respecto a la instalación de software, admite los paquetes con extensión .pet correspondiente a la distribución Puppy Linux, como así soporte para aplicaciones con la extensión .sfs (SquashFS(SFS) es un paquete comprimido de sólo lectura que contiene los archivos y una estructura de directorios.) y desde Puppy Packaged Manager (Administrador de paquetes de Puppy) el acceso a los repositorios de Canonical como a los del mismo Puppy Linux.



También pueden descargar archivos compatibles manualmente desde http://distro.ibiblio.org/quirky/pet_packages-precise/. Desde luego podemos encontrar ayuda en el foro de habla hispana de Puppy (<http://www.murga-linux.com/puppy/index.php?f=24&ppage=30&sort=lastpost&order=DESC>), donde podemos descargar Libreoffice, VLC y muchas otras aplicaciones como así documentación para configuraciones de todo tipo.



Puppy-es Night beta7



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

Conclusión

Muchos catalogan a Puppy Linux como una distribución para equipos obsoletos y muy viejos, y hay parte de razón en estos dichos, pero también hay que tener en cuenta que es una excelente plataforma de trabajo, muy potente y rápida que se adecúa a los procesadores y memorias más recientes; igualmente uno de los puntos más importantes para mí es poder resucitar dispositivos/memorias USB de 512kb, 1gb, 2gb que son prácticamente obsoletos hoy en día, los cuales gracias al desarrollo de este tipo de “distros” y en especial a Puppy Linux y este “puplet” españolizado, se encuentran muy lejos de formar parte de algún museo tecnológico. Destaco especialmente que aunque le falta traducción de algunos menús y software, el esfuerzo por mejorar y llevarlo lo más que se pueda al idioma de Cervantes es muy destacable. Y si uno lo desea puede acercarse a la página de Puppy-es: <http://puppyes.com.ar/> y aparte de descargarlo ¡dar una mano!, una costumbre de la gente de Software Libre en todo el planeta. Un saludo y hasta la próxima!



Es difícil hacer una comparación entre cosas que parecen no tener mucho que ver entre sí. Pero de vez en cuando las cosas se equilibran en algún punto para poder hacerlo. Cuando se mira el mercado del hardware se ven muchos dispositivos informáticos y sus posibilidades de ampliación de funciones ya sea por programación o por accesorios. Ahora, hay casos que parecen demasiado descabellados como comparar un elefante con una aspirina.

Tomemos una serie de teléfonos más emblemáticos desde el 2007, el iPhone. Cuando salió y a pesar de cosas que parecían salidas de apuro. (¿A quién se le ocurrió no incluir copiar y pegar?), era un concepto muy revolucionario, casi parecía tener un futuro de ampliación ilimitado para sus futuras versiones. En la práctica algunas restricciones de programación y de diseño lo han limitado y mucho. Si se mira la cantidad de dispositivos que se han agregado y ve una cantidad limitada de los mismos.

Si, ya se, se han agregado cosas como una cámara frontal, un magnetómetro, se experimentó con algo sin el teléfono. (iPod touch o un tablet de 3.5"). una tablet de 10" y 7,9". Pero en dispositivos externos hay pocas cosas, teclados, parlantes, algunos medidores biométricos, baterías externas, cargadores inalámbricos y algunos conectores para memorias y USB. En algunos casos vemos esos dispositivos y se encuentra que ni en el exterior ni en los manuales o folletos que lo acompañan hay una dirección de email y sitio web. Una forma de evadir el pago de las licencias de desarrollo o no quieren reconocer que han hecho esos dispositivos. En realidad es porque Apple debe aprobar los dispositivos y algunos dispositivos no aprobarían los requerimientos o caprichos de la empresa.

Uno de esos caprichos, por ejemplo, es no permitirle a un desarrollador que use el botón de volumen para disparar la cámara de fotos. Cosa que la empresa implementó por su cuenta después.

Al querer compararlo, uno piensa en cosas como Android o BlackBerry, pero no, la mayoría de ellos tienen casi la misma cantidad de restricciones que los productos de Apple. La comparación va contra Arduino.

Arduino es un proyecto que empezó en 2005 pero que se afianzó en 2007, una pequeña placa de Hardware y Software abierto. Y que fue concebida para ser usada, ampliada y hasta duplicada sin problemas o hacerse uno por cuenta propia. Al



Comparando un elefante con una aspirina

POR Claudio De Brasi

principio se hacían cosas simples como destellar un led o hacer comparaciones de valores analógicos, cosas que se pueden hacer con un 555, 741 o con transistores comunes. Luego se empezaron a hacer cosas más complejas, desde sistemas automáticos pequeños hasta robótica básica. Luego más complejos hasta llegar a cosas como un celular o hasta un satélite. Aquí no hay quien apruebe o limite lo que se puede hacer con el hardware, es más, lo alientan. Si alguien se pone a contar cuántos accesorios e implementaciones de dispositivos hay para un Arduino en <http://www.instructables.com> uno se pierde o deja de contar luego de cierta cantidad. (Yo dejé de contar a los 400), pero van por más de 1500 y en aumento.

A primera vista uno piensa que el sistema más potente es el más versátil pero en realidad no es así. El más versátil siempre es el más ampliable, el más adaptable y por supuesto, el menos restringido.

Los límites de lo ampliable de un dispositivo pueden subsanarse con algún conector, de hecho el mismo Arduino puede servir para subsanar los problemas de falta de conexiones externas de un Celular o Tablet. (Algo que aumenta la versatilidad de ambos dispositivos). La mayor limitante es la licencia, una de Copyright normal que no te deja modificar nada del hardware y otra abierta donde te incita a hacer todo lo que quieras.

Ahora hay nuevos dispositivos. Raspberry Pi hace que con poco dinero tengamos un computador respetable, y pronto Parallella hará que tengas un Supercomputador paralelo. Si uno preside de las campañas de marketing ve que el hardware abierto empieza a abrirse un camino. Y no sólo eso, demuestra que por ser más pequeño o de menores características no es un limitante en sí. A veces el límite está en otro lado y es un error subestimarlos.

Aún recuerdo que una vez durante los años 90 un grupo de radioaficionados hicieron un satélite con una PC XT. Yo me pregunté ¿Por qué no usar una 386?, y la respuesta fue que era más barata y consumía menos y casi igual de versátil. En otras palabras era más apta para la tarea a realizar.

Claudio De Brasi @doldraug

Guía de OpenOffice Calc

POR Rafael Murillo



Como ya lo había dicho en el número anterior, en esta ocasión toca hablar de “**Calc**” (en la nota pasada hablamos un poco sobre el editor de textos de esta suite de herramientas ofimáticas).

Calc es una hoja de cálculo similar a Microsoft Excel (es momento de ir aprendiendo que se llaman así, “hojas de cálculo”, porque es muy molesto que hasta tu jefe te pida “haz un Excel con varias gráficas...”, lo correcto siempre será decir “haz una hoja de cálculo con varias gráficas”), con un rango de características más o menos equivalente. Su tamaño, es decir, la cantidad de memoria que ocupa en nuestros discos duros, es mucho menor y proporciona un número de características no presentes en Excel, incluyendo un sistema que automáticamente define series para representar gráficamente basado en la disposición de los datos del usuario. Calc también es capaz de exportar hojas de cálculo como archivos PDF (aunque ahora ya lo hace también Excel, esta es una característica que OpenOffice.org tiene desde hace ya algún tiempo, obviamente, antes de que Microsoft lo agregara a Excel), cuenta con filtros, auto filtros y puede hacer agrupaciones en tablas dinámicas.

Ahora bien, desde el punto de vista de la programación de macros, Calc no es compatible con el modelo de objetos de Excel (aunque sí con prácticamente todo su repertorio de funciones BASIC), no es tan vulnerable a los virus de macros (o macrovirus) como el producto de Microsoft (Ojo, que dije no es “tan” vulnerable, lo que implica que sí lo es).

Existe la creencia de que, como OpenOffice.org no es tan “comercial”, es decir, está presente en un pequeño porcentaje de equipos en comparación con Microsoft Office, los creadores de virus no tendrán

interés en atacarnos... pues bien, es el peor error que podemos cometer, confiarnos y pensar de esa manera... todo programa puede ser aprovechado para recibir algún ataque, así que no se confíen.

Volviendo a este pequeño resumen, Calc puede abrir y guardar las hojas de cálculo en el formato de archivos de Microsoft Excel (incluso en los más recientes como Office 2010). De hecho, si eres un usuario que está “casado” con el formato de Microsoft, puedes hacer que, el formato predeterminado con el que se guarden tus archivos sea precisamente ese... o bien, continuar usando el formato Open Document Format (ODF). Calc también apoya una amplia gama de otros formatos, tanto para abrir y guardar archivos.

La verdadera “magia”, por decirlo de alguna manera, de OpenOffice.org es que se trata de software libre, por lo cual, cualquier persona puede hacerle mejoras y luego compartirlas con quien quiera. A medida que continúa creciendo la base de usuarios y desarrolladores de OpenOffice.org, las características y mejoras pueden ser agregadas al paquete original a mayor ritmo.

En algunos casos, Calc carece de asistentes para acceder a ciertas características avanzadas asociadas a productos de la competencia como capacidades estadísticas, como el soporte de la barra de error en los gráficos, y el análisis de regresión polinómico, sin embargo esos cálculos se pueden realizar incorporando manualmente las funciones y las relaciones más algunos macros. Otra aplicación FOSS (Free an Open Source Software) llamada Gnumeric proporcionaría un acceso más fácil en estas características del análisis estadístico exponiéndolas a los usuarios con asistentes.

Ahora bien, respecto a la interfaz gráfica, no hay mucho que decir, como pueden ver en la imagen, cuenta con una interfaz similar (prácticamente la misma) que Microsoft Office 2003 (ya que a partir de la versión 2007, Microsoft decidió incorporar “ribbon” a su interfaz de usuario, según ellos para hacerlo más eficiente... aunque lo cierto es que deja mucho que desear (es mi humilde opinión).

- Tiene la clásica barra de título, donde veremos el nombre del archivo en el que estamos trabajando, así como el nombre de la aplicación que lo abre.
- Debajo se encuentra la barra de menús, en los cuales encontraremos todas las funciones, herramientas y opciones del programa.
- Enseguida viene una barra de herramientas y una barra de edición, para el formato del texto, sombreados, bordes, ortografía, crear gráficos, imprimir, etc.
- A continuación viene la barra de fórmulas, desde donde precisamente podemos crear y editar las fórmulas con las que trabajaremos en la hoja de cálculo.
- Respecto a dónde podemos introducir los datos, como podemos ver, es igual que en Excel, existen columnas, filas y celdas, numeradas de la misma manera que en Excel.
- Abajo vamos a encontrarnos con la barra de pestañas o “tabs”, que al igual que en Excel, al principio contamos únicamente con 3 de ellas, pero podemos agregar o quitar las que necesitemos, así como editar sus nombres.
- Finalmente tenemos una barra de estado de la aplicación.

La mayor diferencia que podrán notar en Calc, respecto a Excel, serán los íconos, ya que aquí, podrán tener los que vienen predeterminados, o bien, en algunas distribuciones de Linux (les mentiría si

les digo que en todas porque no he probado con versiones que no estén basadas en Debian) pueden cambiar el paquete de íconos para poner otros que alguien más haya realizado, o bien, realizar sus propios íconos con Gimp, por ejemplo.

Recordemos que esta pequeña Guía va comenzando como un vistazo a la suite de OpenOffice.org, y posteriormete veremos algunas de las acciones quizás más avanzadas, ya que cosas como dar formato al texto, es decir, poner negritas, cursiva, subrayado, etc... es algo muy sencillo y no le costará trabajo a nadie descubrirlo por sí mismos.

Esperamos sus comentarios sobre esta guía y más que nada, sus sugerencias, porque nos ayudarán a darnos cuenta si vamos por buen camino y además, sabremos sus inquietudes y podremos ofrecerles una guía de mejor calidad.



Rafael Murillo
linxack@gmail.com
@linxack



Nos encontramos en la segunda parte de esta guía inicial sobre el uso de KVM, en esta oportunidad vamos a ver algunos aspectos un poco más específicos algunos de ellos relacionados con la

Virtualizando con KVM

Guía inicial parte II

Por Rino
Rondan

línea de comandos. En la nota anterior vimos prácticamente todo desde la gráfica, como, por ejemplo, crear máquinas de una forma rápida y sencilla sin adentrarnos en cuestiones más técnicas, ahora vamos a ver un poco más acerca de la configuración de nuestras máquinas y cómo crearlas pasando también por otras cuestiones sin complicarla demasiado.

Archivos de configuración

Creación de pool de disco

Preparando directorio y pasando imagen:

```
Mkdir /home/crond1/KVM
```

```
cp /home/crond1/download/fedora.iso  
/home/crond1/KVM
```

Creando pool de disco:

```
mkdir /etc/libvirt/storage
```

```
vi /etc/libvirt/storage/disk01.xml
```

El contenido:

```
[restauracion@/home/crond1/KVM $] sudo cat  
/etc/libvirt/storage/disk01.xml
```

```
<pool type='dir'>  
  <name>disk01</name>  
  <capacity>0</capacity>  
  <allocation>0</allocation>  
  <available>0</available>  
  <source>  
  </source>  
  <target>  
    <path>/home/crond1/KVM</path>  
    <permissions>  
      <mode>0700</mode>  
      <owner>-1</owner>  
      <group>-1</group>  
    </permissions>  
  </target>  
</pool>
```

Alguno de los tipos de formato de volúmenes que soporta:

- raw: a plain file
- bochs: Bochs disk image format
- cloop: compressed loopback disk image format
- cow: User Mode Linux disk image format
- dmg: Mac disk image format

- iso: CDROM disk image format
- qcow: QEMU v1 disk image format
- qcow2: QEMU v2 disk image format
- qed: QEMU Enhanced Disk image format
- vmdk: VMWare disk image format
- vpc: VirtualPC disk image format

Definimos el pool de disco basándonos en esa configuración:

```
[restauracion@/home/crond1/KVM $] sudo virsh  
pool-define /etc/libvirt/storage/disk01.xml
```

```
Pool disk01 defined from  
/etc/libvirt/storage/disk01.xml
```

```
[20:15:36]
```

```
[restauracion@/home/crond1/KVM $]
```

Arrancamos el disco

```
[restauracion@/home/crond1/KVM $] sudo virsh  
pool-start disk01
```

```
Pool disk01 started
```

```
[20:16:21]
```

```
[restauracion@/home/crond1/KVM $]
```

Seteamos para que arranque solo

```
[restauracion@/home/crond1/KVM $] sudo virsh  
pool-autostart disk01
```

```
Pool disk01 marked as autostarted
```

```
[20:16:50]
```

```
[restauracion@/home/crond1/KVM $]
```

Listamos los pooles que tenemos:

```
[restauracion@/home/crond1/KVM $] sudo virsh  
pool-list
```

Name	State	Autostart

boot-scratch	active	yes
default	active	yes


```
disk01          active   yes
gnome-boxes     active   no
[20:17:08]
[restauracion@/home/crond1/KVM $]
```

En este caso ya tenía otros pooles creados.

Ver caracterísitcas del disco:

```
[restauracion@/home/crond1/KVM $] sudo virsh
pool-info disk01
Name:          disk01
UUID:          0f123b0c-21a4-ee4b-fc95-
58da8121bbf8
State:         running
Persistent:    yes
Autostart:     yes
Capacity:      249.95 GiB
Allocation:    229.99 GiB
Available:     19.96 GiB
```

```
[20:24:40]
[restauracion@/home/crond1/KVM $]
```

Al no ponerle ningún límite al disco nos toma el espacio que tiene nuestro filesystem.

Creación de disco para la VM

En el apartado anterior creamos el pool de disco para luego ir poniendo nuestras imágenes de disco ahí, el tema radica en decidir la mejor forma de utilizar los tipos de discos.

- Factor de crecimiento:

Cuánto va a crecer nuestra VM, cuánto espacio va a necesitar para la instalación, se podría luego expandir nuestra vm con virt-resize.

Podríamos definir que con una instalación mínima con 2Gb estaríamos bien, si optamos por una típica con 8GB y si vamos por Windows con 16GB posiblemente.

- Tipo de almacenamiento:

podríamos utilizar discos desde un archivo con los diferentes tipos que soporta KVM, cada uno con sus limitaciones. También podríamos utilizar lvm, o también ambas opciones; pero ya sobre una lun o directamente darle un disco físico, y así podemos complicarla mucho más, pero por ahora esto es orientativo.

* Alocación del espacio:

podríamos utilizar un disco con un tamaño variable y que vaya creciendo a medida que se necesita o darle todo el espacio de una.

- Raw o qcow:

Estos son dos tipos de imagen que podemos usar dentro del pool que creamos ya sea la clásica raw image o algo más moderno y complejo que es qcow2 donde podemos hacer que crezca bajo demanda (parecido a sparse files), utilizar snapshot, discos incrementales, etc. Wiki QCOW2

- Snapshot:

Si utilizamos lvm, qcow2 o san, podríamos clonar nuestra máquina y luego utilizar snapshots de éstas.

- Storage pools:

este punto tiene que ver con lo que creamos al principio teniendo en cuenta que podemos hacer pooles configurando tipos de discos. Guía avanzada

Creación de un disco totalmente alocado:

```
[restauracion@/home/crond1 $] dd if=/dev/zero
of=/home/crond1/KVM/guest.img bs=1M
count=2048
2048+0 records in
2048+0 records out
2147483648 bytes (2.1 GB) copied, 19.6027 s, 110
MB/s
```

[22:33:23]

[restauracion@/home/crond1 \$]

Alternativa:

[restauracion@/home/crond1 \$] fallocate -l 8192M
/home/crond1/KVM/guest.img

[22:34:45]

[restauracion@/home/crond1 \$] ls -l
/home/crond1/KVM/guest.img

-rw-rw-r-- 1 crond1 crond1 8589934592 May 1
22:33 /home/crond1/KVM/guest.img

[22:34:54]

[restauracion@/home/crond1 \$]

Creación de un disco pre-allocado

Ambos ejemplos son raw.

[restauracion@/home/crond1 \$] rm -rf
/home/crond1/KVM/guest.img

[22:35:34]

[restauracion@/home/crond1 \$] truncate --
size=2048 /home/crond1/KVM/guest.img

[22:35:51]

[restauracion@/home/crond1 \$] ls -lh
/home/crond1/KVM/guest.img

-rw-rw-r-- 1 crond1 crond1 2.0K May 1 22:35
/home/crond1/KVM/guest.img

[22:35:56]

[restauracion@/home/crond1 \$]

Qcow2:

[restauracion@/home/crond1 \$] qemu-img create -f
qcow2 /home/crond1/KVM/guest2.qcow2 8192

Formatting '/home/crond1/KVM/guest2.qcow2',
fmt=qcow2 size=8192 encryption=off
cluster_size=65536 lazy_refcounts=off

[22:45:59]

[restauracion@/home/crond1 \$]

Para obtener información acerca de ese disco:

[restauracion@/home/crond1 \$] qemu-img info
/home/crond1/KVM/guest2.qcow2

image: /home/crond1/KVM/guest2.qcow2

file format: qcow2

virtual size: 8.0K (8192 bytes)

disk size: 136K

cluster_size: 65536

[22:47:06]

[restauracion@/home/crond1 \$]

El siguiente paso sería usar ese disco que creamos con qcow2 para hacer una instalación limpia y luego clonarlo de manera incremental para que todos los cambios que se generen a partir de ahora se escriban en el disco nuevo, entonces de esta forma tenemos un template limpio siempre para arrancar una nueva instalación. Una vez instalada, la vm con el template no debería ser necesario volver a encenderla.

[restauracion@/home/crond1 \$] qemu-img create -b
/home/crond1/KVM/guest2.qcow2 -f qcow2 guest2-
clone.qcow2

Formatting 'guest2-clone.qcow2', fmt=qcow2
size=8192

backing_file='/home/crond1/KVM/guest2.qcow2'
encryption=off cluster_size=65536
lazy_refcounts=off

[22:53:15]

[restauracion@/home/crond1 \$] qemu-img info
guest2-clone.qcow2

image: guest2-clone.qcow2

file format: qcow2

virtual size: 8.0K (8192 bytes)

disk size: 196K

cluster_size: 65536

backing file: /home/crond1/KVM/guest2.qcow2

[22:53:40]

[restauracion@/home/crond1 \$]

Como verán la salida de información del disco que creamos, marca que proviene de otro archivo, pero a partir de ahora sólo va a escribir en este último.

Acá en ambos casos los discos los creamos pero no los usamos.

Instalación de VM's

Ahora que ya tenemos definidos los discos, podemos empezar a crear las VM's.

¡¡¡Agárrense porque vamos por la línea de comando!!!

```
[restauracion@/home/crond1 $] virt-install -n www
-r 1024 -f /home/crond1/KVM/guest2.qcow2 -s 30 --
vcpus=1 --os-type linux --os-variant=fedora18 --
network bridge=virbr0 --nographics --
location='http://fedora.gtdinternet.com/releases/test/
19-Alpha/Fedora/x86_64/os/' --extra-
args='console=tty0 console=ttyS0,115200n8 serial'
```

Starting install...

Retrieving file .treeinfo...
/ 2.2 kB 00:00:00 !!!

Retrieving file vmlinuz...
/ 11 MB 00:00:45 !!!

Retrieving file initrd.img...
/ 59 MB 00:04:10 !!!

Creating domain...
/ 0 B 00:00:01

Connected to domain www

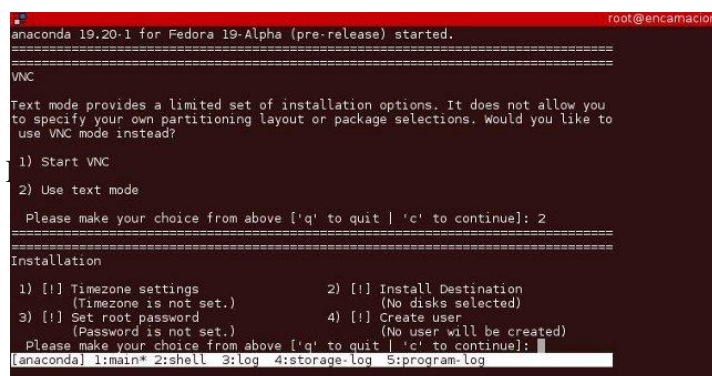
Escape character is ^]

```
[ 0.000000] Initializing cgroup subsys cpuset
[ 0.000000] Initializing cgroup subsys cpu
[ 0.000000] Linux version 3.9.0-
0.rc6.git2.3.fc19.x86_64 (mockbuild@bkernel02)
(gcc version 4.8.0 20130322 (Red Hat 4.8.0-1)
(GCC) ) #1 SMP Mon Apr 15 20:34:53 UTC 2013
[ 0.000000] Command line:
method=http://fedora.gtdinternet.com/releases/test/1
9-Alpha/Fedora/x86_64/os// console=tty0
console=ttyS0,115200n8 serial
[ 0.000000] e820: BIOS-provided physical RAM
```

map:

```
[ 0.000000] BIOS-e820: [mem
0x0000000000000000-0x000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem
0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem
0x00000000000f0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem
0x00000000000100000-0x00000000003ffffdfff] usable
[ 0.000000] BIOS-e820: [mem
0x00000000003fffe000-0x00000000003ffffff] reserved
[ 0.000000] BIOS-e820: [mem
0x0000000000feffc000-0x0000000000feffffff] reserved
[ 0.000000] BIOS-e820: [mem
0x0000000000fffc0000-0x0000000000ffffffff] reserved
[ 0.000000] NX (Execute Disable) protection:
active
```

Luego de un rato tenemos algo así:



Como verán le pasamos varios flags, el proceso sigue pero lo cortamos.

virt-install \ ---> **comando para instalar**

-n www \ ---> **nombre de la vm**

-r 1024 \ ---> **memoria que se le va asignar**

-f /home/crond1/KVM/guest2.qcow2 \ ---> **dónde está el disco**

-s 30 \ ---> **espacio de disco**

--vcpus=1 \ --> **procesadores**

--os-type linux \ --> **tipo de GuestOs**

--os-variant=fedora18 \ --> **variante de Linux**

--network bridge=virbr0 \ --> **qué placa de red usa**

--nographics \ --> **Que no utilice la gráfica**

--

location='http://fedora.gtdinternet.com/releases/test/19-Alpha/Fedora/x86_64/os/' \ --> **de dónde saca la imagen para instalar**

--extra-args='console=tty0 console=ttyS0,115200n8 serial' ---> **al elegir que no use gráfica habilitamos la terminal para ver la instalación**

Ahora vamos a ver un ejemplo para arrancar desde una ISO y usar el protocolo spice que es el que reemplaza a VNC.

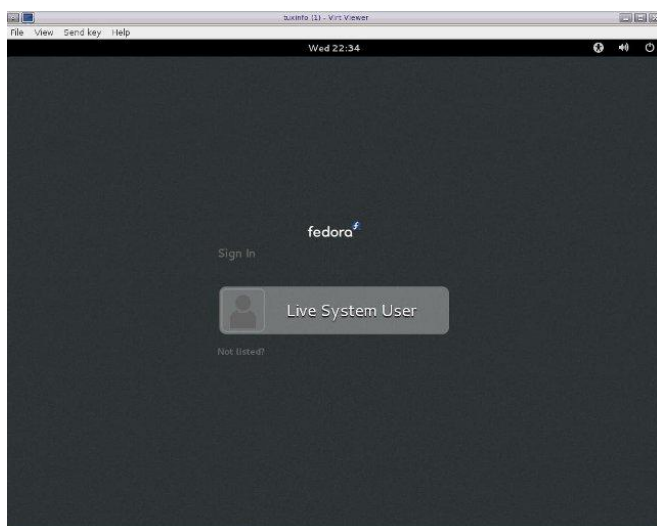
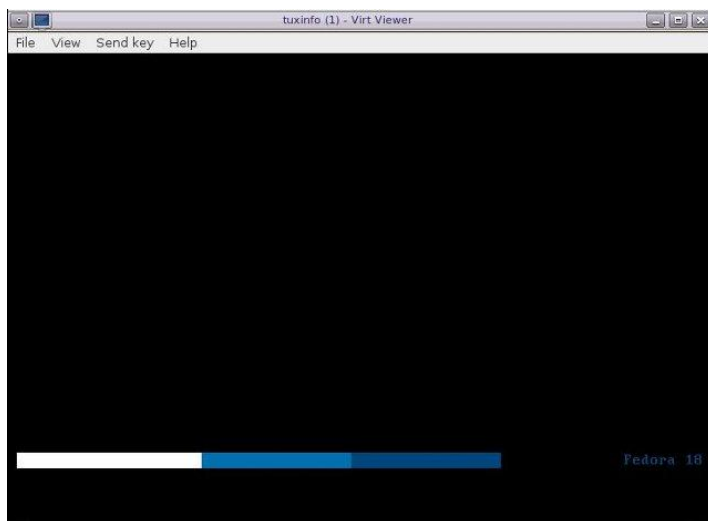
```
[restauracion@/home/crond1/KVM $] virt-install -n  
tuxinfo -r 512 --vcpus=1 --disk  
path=/home/crond1/KVM/f18.qcow2 --cdrom  
/home/crond1/KVM/f18.iso --network  
bridge=virbr0 --graphics  
spice,port=5930,listen=0.0.0.0
```

Starting install...

Creating domain...
/ 0 B 00:00:00

(virt-viewer:24513): GSpice-WARNING **:
Warning no automount-inhibiting implementation
available

Luego de esto, aparecerá la ventana de spice.



La cantidad de opciones que tiene el virt-install es enorme:

```
[restauracion@/home/crond1 $] virt-install --help
```

Usage: virt-install --name NAME --ram RAM
STORAGE INSTALL [options]

Options:

--version show program's version number
and exit

-h, --help show this help message and exit

--connect=URI Connect to hypervisor with
libvirt URI

General Options:

-n NAME, --name=NAME
Name of the guest instance

-r MEMORY, --ram=MEMORY
Memory to allocate for guest instance
in megabytes

--vcpus=VCPUS Number of vcpus to
configure for your guest. Ex:

--vcpus 5

--vcpus 5,maxcpus=10

--vcpus sockets=2,cores=4,threads=2

--cpuset=CPUSET Set which physical CPUs
domain can use.

--cpu=CPU CPU model and features. Ex: --
cpu coreduo,+x2apic

`--description=DESCRIPTION`
Human readable description of the VM to store in the generated XML.

`--security=SECURITY`
Set domain security driver configuration.

`--numatune=NUMATUNE`
Tune NUMA policy for the domain process.

Installation Method Options:

`-c CDROM, --cdrom=CDROM`
CD-ROM installation media

`-l LOCATION, --location=LOCATION`
Installation source (eg, nfs:host:/path, http://host/path, ftp://host/path)

`--pxe`
Boot from the network using the PXE protocol

`--import`
Build guest around an existing disk image

`--init=INIT`
Path to init binary for container guest. Ex:

`--init /path/to/app` *(to contain an application)*

`--init /sbin/init` *(for a full OS container)*

`--livecd`
Treat the CD-ROM media as a Live CD

`-x EXTRA, --extra-args=EXTRA`
Additional arguments to pass to the install kernel

`--initrd-inject=INITRD_INJECTIONS`
Add given file to root of initrd from --location

`--os-type=DISTRO_TYPE`
The OS type being installed, e.g. 'linux', 'unix', 'windows'

`--os-variant=DISTRO_VARIANT`
The OS variant being installed guests, e.g. 'fedora6', 'rhel5', 'solaris10', 'win2k'

`--boot=BOOTOPTS` *Optionally configure post-install boot order, menu, permanent kernel boot, etc.*

Storage Configuration:

`--disk=DISKOPTS` *Specify storage with various options. Ex.*

`--disk path=/my/existing/disk`

`--disk path=/my/new/disk,size=5` *(in gigabytes)*

`--disk vol=poolname/volname,device=cdrom,bus=scsi,...`

`--nodisks` *Don't set up any disks for the guest.*

`--filesystem=FILESYSTEMS`
Pass host directory to the guest. Ex:

`--filesystem /my/source/dir,/dir/in/guest`

`--filesystem template_name/,type=template`

Networking Configuration:

`-w NETWORK, --network=NETWORK`
Configure a guest network interface. Ex:

`--network bridge=mybr0`

`--network network=my_libvirt_virtual_net`

`--network network=mynet,model=virtio,mac=00:11...`

`--nonetworks` *Don't create network interfaces for the guest.*

Graphics Configuration:

`--graphics=GRAPHICS`
Configure guest display settings. Ex:

`--graphics vnc`

```
--graphics
spice,port=5901,tlsport=5902
--graphics none
--graphics
vnc,password=foobar,port=5910,keymap=ja
--noautoconsole  Don't automatically try to
connect to the guest
```

```
console
```

Device Options:

```
--controller=CONTROLLER
```

Configure a guest controller device.

Ex:

```
--controller type=usb,model=ich9-
```

ehci1

```
--serial=SERIALS  Configure a guest serial
device
```

```
--parallel=PARALLELS
```

Configure a guest parallel device

```
--channel=CHANNELS  Configure a guest
communication channel
```

```
--console=CONSOLES  Configure a text console
connection between the guest
```

and host

```
--host-device=HOSTDEVS
```

Configure physical host devices
attached to the guest

```
--soundhw=SOUNDHW  Configure guest sound
device emulation
```

```
--watchdog=WATCHDOG
```

Configure a guest watchdog device

```
--video=VIDEO  Configure guest video
hardware.
```

```
--smartcard=SMARTCARD
```

Configure a guest smartcard device.

Ex:

```
--smartcard mode=passthrough
```

```
--redirdev=REDIRDEV
```

Configure a guest redirection device.

Ex:

```
--redirdev
```

```
usb,type=tcp,server=192.168.1.1:4000
```

```
--memballoon=MEMBALLOON
```

Configure a guest memballoon
device. Ex:

```
--memballoon model=virtio
```

Virtualization Platform Options:

```
-v, --hvm  This guest should be a fully
virtualized guest
```

```
-p, --paravirt  This guest should be a
paravirtualized guest
```

```
--container  This guest should be a container
guest
```

```
--virt-type=HV_TYPE
```

Hypervisor name to use (kvm, qemu,
xen, ...)

```
--arch=ARCH  The CPU architecture to
simulate
```

```
--machine=MACHINE  The machine type to
emulate
```

```
--noapic  Disables APIC for fully
virtualized guest (overrides
```

value in os-type/os-variant db)

```
--noacpi  Disables ACPI for fully
virtualized guest (overrides
```

value in os-type/os-variant db)

```
-u UUID, --uuid=UUID
```

UUID for the guest.

Miscellaneous Options:

```
--autostart  Have domain autostart on host
boot up.
```

```
--print-xml  Print the generated domain XML
rather than define the
guest.
```

```
--print-step=XMLSTEP
```

Print XML of a specific install step
(1, 2, 3, all)

rather than define the guest.

```
--noreboot  Don't boot guest after completing
install.
```


--wait=WAIT Time to wait (in minutes)
 --dry-run Run through install process, but
 do not create devices
 or define the guest.
 --force Forces 'yes' for any applicable
 prompts, terminates
 for all others
 -q, --quiet Suppress non-error output
 --prompt Request user input for ambiguous
 situations or
 required options.

-d, --debug Print debugging information
 [23:43:31]
 [restauracion@/home/crond1 \$]

Para ir cerrando todo, podemos ver algunos
 comandos de virsh así luego en la próxima entrega
 seguimos profundizando.

Como vemos, ahí tenemos las dos vm:

[restauracion@/home/crond1 \$] virsh list --all

<i>Id</i>	<i>Name</i>	<i>State</i>

2	www	running
3	tuxinfo	running

[00:32:50]
 [restauracion@/home/crond1 \$]

Si quisieras eliminarlas:

[restauracion@/home/crond1 \$] virsh destroy www
 Domain www destroyed

[00:33:22]
 [restauracion@/home/crond1 \$] virsh list --all

<i>Id</i>	<i>Name</i>	<i>State</i>

3	tuxinfo	running
-	www	shut off

[00:33:25]
 [restauracion@/home/crond1 \$]

 Primero el destroy para apagarla de forma abrupta,
 luego:

 [restauracion@/home/crond1 \$] virsh undefine www

 Domain www has been undefined

[00:34:13]
 [restauracion@/home/crond1 \$] virsh list --all

<i>Id</i>	<i>Name</i>	<i>State</i>

3	tuxinfo	running

[00:34:56]
 [restauracion@/home/crond1 \$]

Luego para arrancar una vm:

[restauracion@/home/crond1/KVM \$] virsh start
 tuxinfo
 Domain tuxinfo started

[00:36:14]
 [restauracion@/home/crond1/KVM \$]

Apagar:

[restauracion@/home/crond1/KVM \$] virsh
 shutdown tuxinfo
 Domain tuxinfo is being shutdown

[00:37:24]
 [restauracion@/home/crond1/KVM \$]

Si quisiéramos entrar por spice deberíamos ejecutar el comando como vimos en la entrega anterior.

```
[restauracion@/home/crond1/KVM $] spicec -h  
localhost --port 5930
```

Esto es todo, también si algo falla por acá, pueden usar la gráfica, pero todavía quedan muchas cosas más por ver.

Referencias : <http://virt-tools.org/learning/install-with-command-line/>

http://www.server-world.info/en/note?os=Fedora_18&p=kvm&f=2

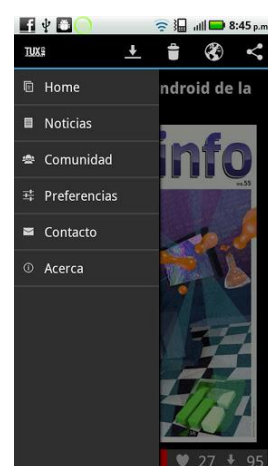
<http://www.linux-kvm.com/content/how-you-can-use->

gemukvm-base-images-be-more-productive-part-1



Rino Rondon
Ambassador Fedora
RHCEPIC2

ADELANTO DE TUXINFO 59 INCLUIREMOS UN INFORME COMPLETO DEL SMARTPHONE NEXUS 4



TUX **INFO**
WWW.TUXINFO.COM.AR