

cotej



Quando eramos
libres...



ENTREVISTAS

Jorge Castellanos
&
Hector A. Mantellini
Charlan con nosotros

ANDROID
(APK)
DESDE
GNU/LINUX



Android
su historia



SE LAS JUEGA
EN SUS
CAMBIOS

Curso



Parte VI

OpenLDAP
En aplicaciones web



PHPMONITOR

Linux Mint
La distro del
usuario final



Staff:

Juan C. Karroum (JCK)
Héctor A. Mantellini (Xombra)

Colaboradores:

Naudy Villaroel
Angel Cruz (abr4xas)
José Parella (bureado)
AWVEN
Alvaro Hernandez (ajha)
David Rondon (emerling)
Juan Moreno (apostols)

*Todo el contenido está bajo
licencia de Creative Commons.*

*Puede copiar, distribuir, mostrar
públicamente su contenido y hacer
obras
derivadas, siempre y cuando:*

- a) Reconozca los créditos de la obra*
- b) No use de forma comercial*
- c) Comparta bajo la misma licencia.*



@vaslibre 

 vaslibre

Herramientas usadas:

Inkscape 0.48.1

Scribus 1.3.9

EDITORIAL

VaSlibre al tener ya 6 años como grupo de Usuarios de Software Libre ha madurado, demostrando que cuando las cosas se hacen bien desde un principio se logran los objetivos planteados. Cotejo cumple este mes su primer año de vida, esperemos que cumpla muchos.

Este año ha sido el año de VaSlibre como grupo, se han sumado a sus filas un número significativo de nuevos miembros aportando su grano de arena, igualmente ha sido el periodo en el que más eventos se han hecho y en el que mayor participación ha habido. Hemos logrado llegar a más de 1500 personas en estos eventos en la difusión de las bondades del Software Libre.

Importantes instituciones educativas (INCES Carabobo, UNA Local Carabobo) nos han ayudado en nuestra labor. Gracias una vez más por apoyarnos.

Estamos en conversaciones con dos Liceos de la región para que participen activamente el año entrante.

SafeCreative.org también nos ha apoyado, otorgando información y material para el uso de Licencias, brindando de esta manera un piso jurídico para el registro de nuestros trabajos.

Cotejo ha sido acogida por la comunidad del software libre, muchos de los sitios y blog referentes a Linux de habla hispana han colaborado de una u otra forma en hacerla llegar a miles de internautas. Al momento de escribir estas líneas la edición 6 ya lleva mas de 13 mil lecturas/descargas y las otras ediciones rondan números similares.

Esperamos con el aporte de todos ustedes seguir en esta encomiable labor.

VaSlibre les desea una Feliz Navidad y un Venturoso año nuevo 2012.

Sé Libre usa GNU/Linux!



CONTENIDO

- Entrevista a:
 - Jorge Castellano (04)
 - Hector A. Mantellini (06)
- Android... su historia (09)
- Curso PHP Parte IV (11)
- Cuando eramos libres... (14)
- Descargando aplicaciones Android (APK) (16)
- gtkOrphan chao huérfanos (19)
- LDAP y aplicaciones Web (20)
- Ubuntu se las juega en sus cambios (24)
- Linux Mint: La distro del usuario final (26)
- Sitios recomendados (29)
- Plan Ceibal (30)
- BleachBit el limpiador (33)
- Eventos VaSlibre (34)
- PHPmonitor Monitoriza tu servidor(es) (35)
- Estructura de Directorios y/o Carpetas en GNU/Linux (37)
- El Tao de la Programacion (38)



VaSlivre ha realizado sendas entrevistas a dos de sus miembros fundadores: Jorge Castellanos y Hector A. Mantellini.

La primera entrevista fue a el profesor **Jorge Castellanos**, uno de los Fundadores de VaSlivre

1.- Como conociste el Software Libre?

Conocí el Software Libre el año 1995 a través de una revista llamada Informática Fácil Multimedia que traía un CD (que aún conservo) con el Sistema Operativo Linux en la versión Slackware 2.3 con el kernel 1.2.

Después de mucho esfuerzo, bajando los paquetes que faltaban con una internet que trabajaba con modem de 9600 baudios, logré instalarla en una PC 486 de la época.

2.- Como fue tu primera experiencia en el uso del Software Libre?

Mi primera experiencia fue apasionante, porque era como hacer todo desde cero. En el fondo el sistema operativo Linux me parecía similar al Unix que había conocido años atras pero lo interesante era que el esfuerzo que uno le agregaba para que las cosas funcionaran lo hacía sentirse parte del software instalado. Por cierto, como una anécdota le comento que uno de los

atractivos en aquella época era lograr que una vez instalado el Linux se pudiera intalar y ejecutar exitosamente el juego Doom.

3.- Cual ha sido tu distribución favorita y porque?

Mi distro favorita a estado vinculada a distintas etapas de mi vida.

En el año 1995 cuando conocí Linux instalé el Slackware.



Luego, empecé a trabajar en la Universidad de Carabobo, en el Department

o de Computación de FACYT y allí conocí el recién salido Red Hat version 1.0, el cual comparado con Slackware resultaba muy fácil de instalar y configurar.

En el 2000 conocí la famosa Mandrake 7.0 que me cautivó por sus interfaces y configuración. Además funcionaba muy bien e incluía el primer esfuerzo de una suite de oficina muy compatible con el Office de Microsoft, el famoso paquete StarOffice.

Luego con la aparición de Mandriva y los multiples problemas que presentaba esta distro con el hardware que yo utilizaba empecé a usar Debian 3.0 (woody). Aunque al

principio Debian me gusto bastante, los kernel de la época resultaba inusualmente complicados de configurar y adecuarlos al hardware con el cual trabajaba.

Gracias a la influencia de mi amigo Roliverio empecé a usar la distro Gentoo que para mis intereses resultó muy atractiva por las facilidades de configuración y compilación del kernel y su amplia gama de alternativas de configuración y además la posibilidad de trabajar en el mismo ambiente con códigos compilados en 32 y 64 bits.

Hace ya dos años dejé de usar el Gentoo y actualmente tengo instalado el Arch Linux. Arch Linux aunque tiene los paquetes en binario tiene la posibilidad de compilar y crear sus propios paquetes como Gentoo. Además creo que Arch Linux, además de disponer de un excelente wiki, es muy sencilla para configurar porque está compuesta de paquetes recientes y optimizada para las arquitecturas x86 y x86_64 por lo cual me gustó desde que la instalé.

Creo que la distro que uno use se debe ajustar a nuestras necesidades de software y hardware que uno tenga.

Para mi caso actual, necesito mantener mis herramientas de programación actualizadas

usando portátiles con arquitectura x86_64, además necesito instalar paquetes muy recientes en 32 y 64 bits, sin mucho trauma.

4.- Desde cuando perteneces a VaSlibre, cual es tu opinión del grupo?

Desde su fundación una mañana de 24 septiembre de 2005 en el Centro Comercial la Granja. Aunque sigo la lista de correos y mantengo algún contacto personal con mis amigos Roliverio, Ajha y Xombra; no estoy activo en VaSlibre desde hace algún tiempo.

5.- Que opinas del Software Libre en Venezuela?

Opino que se está desarrollando pero quizás no al ritmo que muchos lo creían. En mi trabajo usamos las herramientas de software libre para las asignaturas de programación, es decir, los compiladores GNU C, C++, los editores y por supuesto el sistema operativo GNU/Linux. Pero con pesar he visto que en instituciones del estado aún prefieren Windows tanto para la enseñanza como para la administración.

6.- Estas en algún proyecto actualmente en la comunidad del Software Libre?



Actualmente me dedico a terminar mi tesis doctoral, la cual consiste en el desarrollo de una capa de un software para manejar las necesidades de memoria de una biblioteca de cómputo que resuelve sistemas lineales dispersos. Este desarrollo lo he hecho en lenguaje ANSI C usando los compiladores GNU. Ante la ausencia de herramientas especializadas para el trazado y perfilado de aplicaciones multithread he estado usando software desarrollado por Google y la Universidad de Oregon.

7.- Eres de los puristas y radicales en cuanto al SL (software libre) o eres de los light& Cool?

Me considero light, porque creo que si restrinjo mi trabajo solo a Software Libre siento que me estoy limitando. Como fan de los Sistemas Operativos, además de GNU/Linux uso MacOS y muy eventualmente MS-Windows.

8.- Que opinas de los eventos

que se están haciendo Flisol - CNSL - Encuentro de Comunidades - Cayapas?

Pienso que esos eventos son muy importantes y celebro la acción de todos aquellos que suman esfuerzos para ellos sean una realidad.

Personalmente he trabajado en años anteriores colaborando con los festivales de instalación.

9.- Tienes algún blog, escribes con frecuencia?

Tuve un tiempo en que escribí artículos relacionados con Software Libre en diarios nacionales pero por razones de índole personal lo dejé de hacer.

10.- Eres Fans de Richard Stallman?

No soy fan de RMS aunque en un momento de mi vida asistí a su conferencia y hasta una foto con él me saqué. En todo caso pienso que RMS ha sido un factor muy importante en desarrollo y promoción del Software Libre y comparto muchas de sus posturas aunque no todas.

Gracias Jorge por compartir con nosotros.

A continuación entrevista realizada a **Héctor A. Mantellini (xombra)** uno de los fundadores de Vaslibre.

1.- Como conociste el Software Libre?

Creo que fue por el año 2001, que mi amigo personal Alexander Garzon me habló RedHat, la versión 6.2 (Zoot) de la que habían hecho un artículo en una revista, lamentablemente no recuerdo cual era. Por curiosidad en el tema investigué, y me llamó muchísimo la atención esta filosofía en el uso y desarrollo del software.

En ese tiempo trabajaba con MS-Windows (aún lo uso en ocasiones) y daba clases de su uso, también trabajaba algunas veces con el "Sistema Operativo" de Novell (necesitaba de MS-DOS para montarse), por lo que estaba familiarizado con la forma de aplicar el uso de programas, aunque pensaba que debería haber una forma de poder saber como funcionaban esos programas, "digamos que crackeandolos o haciendoles ingeniería inversa y mejorandolos o ajustarlo a mis necesidades del momento" y poder distribuirlos sin problemas de licencia. Claro, aunque sabemos que en Venezuela, aún el uso de licencias de software esta en pañales. Por ello, esta forma de

trabajo me fascino desde un principio.

2.- Como fue tu primera experiencia en el uso del Software Libre?

Como dije anteriormente RedHat 6.2 me gustó y empecé a usarla, el uso de la consola (mis inicios fueron en cpm80) es genial y aprendí mucho sobre la estructura de los sistemas operativos, drivers, compilación... algo alusinante. Lamentablemente cuando RedHat llegó a la versión 9 pasó a ser de pago y tuve que probar con otras distribuciones, con resultados alentadores y otros no tanto.



En el año 2004 fui al 1er Foro Mundial del Software Libre en Caracas, donde conocí y compartí con otros entusiastas del mundo del Software Libre, y desde ese momento me encamine en serio en la ruta del movimiento.

3.- Cual ha sido tu distribución favorita y porque?

Esto depende de varios factores, empecé con RedHat 6.2 al llegar a la versión 9 pasó a ser de pago, y me cambie a Fedora Core 1 (Yarrow), pero no llenaba mis expectativas.

El amigo Ahja (Alvaro Hernandez) me habló de SlackWare y decidí intartar con ella pero uff! luego de usar algo suave como RedHat y paso a SlackWare fue como tener que aprender todo de cero jajaja y decidí alejarme de ella, luego mi amigo Roliverio me aconsejo usar Gentoo, la cual era algo menos traumatica que Slackware, pero... igual había que darse duro para hacer que funcionara ciertas partes de mi máquina.

Volví a Fedora por un tiempo hasta que en un cd de una revista conocí a Mandrake (actual Mandriva) que era derivada de RedHat y migre a ella con entusiasmo. De allí otro amigo Andy Borrero me presentó a mi Favorita UBUNTU la versión 6.06 LTS "Dapper Drake".



Su facilidad de uso me cautivo de inmediato y la usé hasta la versión 10.10 "Maverick Meerkat". Por razones de uso (no me gusta Unity) ahora uso LinuxMint (derivada de Ubuntu), la cual probé por vez primera la versión 10 "Julia", actualmente uso LinuxMint 11 "Katya" con Gnome3

4.- Desde cuando perteneces a VaSlibre, cual es tu opinión del grupo?

Desde sus inicios en septiembre del 2005, todos los fundadores



venimos de Velug Valencia (creo que en Carabobo ese grupo murio) y en una reunión Alvaro Hernandez (Ajha), Julio C. Ortega (Roliverio), Jorge Castellanos, Juan D'Alessandro, Juan Carlos Karroum, Decio Rodríguez, Luis Mora y yo decidimos desligarnos de Velug Valencia y formar nuestro propio Grupo de Usuarios.

El grupo como tal es bastante compacto y fuerte, hoy día se han sumado a sus filas personas con las mismas ansías que nosotros dándole solidez y diversidad.

Entre las cosas buenas que tiene el grupo, que cuando hacemos un evento, cualquiera de los miembros fundadores da la mano para que este sea llevado a bien.

Como en todos los grupos hay diferencias de criterio y opinión lo cual nutre en forma positiva a VaSlibre.

5.- Que opinas del Software Libre en Venezuela?

Con el dinero que se ha invertido debería ser el mejor del mundo, empero siempre hay intereses personales que socavan lo que se ha venido logrando. En gran parte es culpa de los Grupo ya que si cada uno de nosotros informara de las irregularidades en el manejo de fondos asignados al Software Libre otro gallo cantara.

Otro punto importante es las peleas innecesarias entre grupos. VaSlibre apoya

dentro de sus posibilidades a los diferentes grupos, incluso hemos servido de mediadores para la reconciliación entre grupos.

Un proyecto Genial ha sido la distribución nacional Canaima, pero como indique anteriormente la mezquinda, protagonismo y corrupción ha hecho que el proyecto por momentos parezca detenido.

6.- Estas en algún proyecto actualmente en la comunidad del Software Libre?

He estado dando charlas en diferentes ciudades del territorio nacional, de hecho este año es donde he tenido la mayor participación en el proyecto de divulgación llevado por VaSlibre.

También he puesto mi granito de arena en las traducciones al español en Ubuntu. Desde hace unos meses andamos Jhon Vera (jjedi_ve) y yo en la creación de una "mini-distro" basada en Canaima para la Universidad Nacional Abierta, Local Carabobo.

7.- Eres de los puristas y radicales en cuanto al SL (software libre) o eres de los light& Cool?

Soy light, cada quien usa el sistema operativo que desee y que cumpla con las tareas. Aunque recomiendo a los usuarios noveles y no tan noveles que usen LinuxMint por su facilidad de uso y diversidad de aplicaciones que posee en repositorios.

No hay que cerrarse en que un sistema operativo es libre, medio libre o no libre, es necesario ponerse en los zapatos del usuario final, y brindarle la mejor opción para su máquina.

En ocasiones uso MS-Windows y una que otra MacOS



8.- Que opinas de los eventos que se están haciendo Flisol - CNSL - Encuentro de Comunidades - Cayapas?

Vayamos por parte:

El Flisol en Venezuela es el más importante de los eventos del software libre, aunque han habido "personas" que intentan politilizar la idea. La amiga María "Tatica" Leandro junto con sus colaboradores ha hecho que este evento sea lo que es hoy día en Venezuela. Hasta los momentos he participado de una u otra forma en él en la ciudad de Valencia con el grupo VaSlibre.

El CNSL también es importante, aunque ha tenido sus detractores más que todo por las personas que estan alrededor de Octavio Rossell (tr0n), pero no cabe duda que es un evento necesario para la difusión del software libre en Venezuela.

Con respecto a Encuentro de Comunidades, tuve la oportunidad de asistir como ponente al celebrado en la ciudad de Maracay, este tipo de eventos solo deberían ser

un poco más técnicos debido a que los asistentes ya poseen un nivel en el uso de GNU/Linux.

Las Cayapas son una idea genial, a la última que se celebró por motivos medicos no pude asistir. Son importante porque allí se exponen y se debaten que camino deben tomar y como deben manejarse los diferentes proyectos.

9.- Tienes algún blog, escribes con frecuencia?

Blog no tengo, mi sitio web xombra.com publica uno que otro artículo de seguridad o tips de uso de Ubuntu. Las noticias son aportadas por parte del staff que conforman a xombra.com.

Tambien escribo en Awven.com sobre desarrollo web, SEO, Programación PHP. Y por supuesto en el sitio de VaSlibre.

10.- Eres Fans de Richard Stallman?

En un tiempo si, pero desde hace 3 años no comparto la forma como lleva las ideas del movimiento. Hay que evolucionar. Pienso que RMS es y fue importante en la difusión de la filosofía del Software Libre, y ya es hora que suelte la batuta o en el mejor de los casos se adapte a las nuevas formas y tendencias del uso de la tecnología.

Héctor A. Mantellini
Twitter: @xombra
Identi.ca: xombra
<http://www.xombra.com>

TIPS

Montar manualmente un dispositivo USB

Abrir consola

Escribir

```
sudo mkdir /media/usb
```

Luego

```
sudo fdisk -l
```

Sí queremos montarlo como /dev/sdb1

escribimos:

```
#Montarlo como FAT16 o FAT32
```

```
sudo mount -t vfat /dev/sdb1 /media/usb -o uid=1000,gid=100,utf8,dmask=027,fmask=137
```

```
#montarlo como NTFS
```

```
sudo mount -t ntfs-3g /dev/sdb1 /media/usb
```

Para desmontarlo

```
sudo umount /media/usb
```

ANDROID... SU HISTORIA

La historia hasta ahora... de Android, de lo que sabemos (o quizás no) desde sus inicios por allá en el 2003 hasta los momentos, muchos dirán que es una más del montón... Pero, PROBLEM? xD...

Orientada más que todo a esas personas que aun no conocen la historia desde su nacimiento hasta

TODOS los beneficios que nos brinda tener un equipo con Android y lo que nos falta por saber...

Que es Android?

Android es un sistema operativo basado en el núcleo Linux diseñado originalmente para dispositivos móviles, tales como teléfonos inteligentes, pero que posteriormente se expandió su desarrollo para soportar otros dispositivos tales como tablet, reproductores MP3, netbook, PC, televisores, lectores de e-book e incluso, se han llegado a ver en el CES*, microondas y lavadoras.

"El International Consumer Electronics Show o (CES) (Feria Internacional de Electrónica de Consumo en español) es un evento que se da cada enero en la ciudad de Las Vegas, Nevada, Estados Unidos."

Historia de Android

- 2003: Android Inc. Empresa fundada para desarrollar, dijo el co-fundador Andy Rubin "... dispositivos móviles inteligentes que son más conscientes de la ubicación de su propietario y preferencias"

2005: Google: adquiere android se sabía muy poco acerca de Android Inc., pero esto se supone que este era el plan de Google para entrar en el mercado de la telefonía móvil.

2007: Open Handset Alliance anuncia Android: Un grupo de más de 80 compañías de

tecnología y móviles como HTC, Samsung, Sony y Google, que se han unido para acelerar la innovación en móviles presentará su primer producto: "Android"



2009: Android versión 1.0: El primer dispositivo Android fue el HTC Dream (G1), que incluye una amplia gama de servicios de integración de Google.

2011: Android llega al n° 1 en el smartphone arena: A finales de 2010 dominó el mercado de los EE.UU. (59%) y Europa (46%).

Status

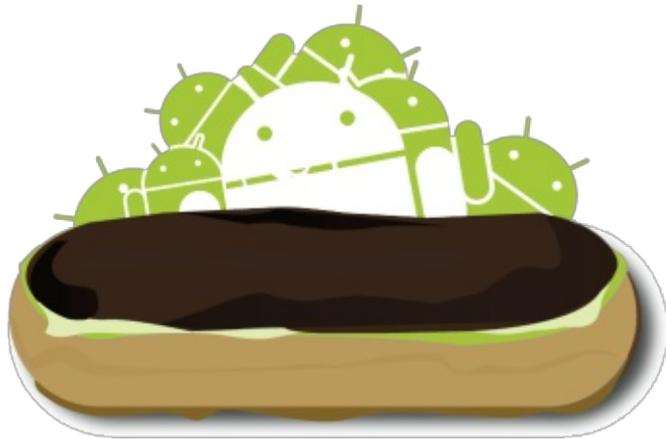
135 MILLONES totales de activaciones... Existen 250.000 Android apps, hay 410 Dispositivos Android, trabaja con 231 Operadoras de Red, esta en 123 países hay registros de 550.000+ activaciones diarias en el mundo...

Si no te emociona tener un equipo con Android, y sigues pensando que tener un BlackBerry es lo mejor que hay pues, te veo mal...



ANDROID... SU HISTORIA

¡Claro! Puedes usar lo que tu quieras pero, nunca tendras la "LIBERTAD" de poder hacer infinitas cosas y disfrutar de +250 mil aplicaciones que existen en el market y



Búsquedas de texto a voz, Búsqueda universal...

> Eclair 2.0: Sincronización de cuenta, Email Exchange, Cámara mejorada...

> Froyo 2.2: Conexión Wifi, Soporte flash, Mejor rendimiento...

> GingerBread 2.3: Soporte Nativo VoIP, Administrador de descargas, Interfaz de usuario más

de Honeycomb. Opción de utilizar los botones virtuales en la interfaz de usuario, en lugar de los botones táctiles capacitivos...

Basado en trabajo de <http://bit.ly/vgFFOB>

Abr4xas
<http://www.abr4xas.org>
Twitter: @abr4xas

aumentar la experiencia al usar un Android. Eso sin mencionar que tu mism@ puedes hacer tus propias aplicaciones con las características que desees...

sencilla y rápida...

> HoneyComb 3.0: Soporte un tabletas, Navegación por pestañas, Video chat...

Evolución de Android (Características)

> Cupcake 1.5: Grabación de vídeo Widgets...

> Donut 1.6: Market,

> Ice Cream Sandwich 3.1: Version que unifica el uso en cualquier dispositivo, tanto en teléfonos, tablets, televisiones, netbooks... Interfaz limpia y moderna con una nueva fuente llamada "Roboto", Muy al estilo

TIPS

Colocar iconos arriba derecha

Abrir consola y escribir:

```
gconftool-2 --type string --set /apps/metacity/general/button_layout "menu:maximize,minimize,close"
```

y listo!!!



Curso PHP Parte IV

Continuamos con el curso de programación en PHP.

Arreglos (tablas o vectores)

Las tablas (o array en inglés), son muy importantes en PHP, ya que generalmente, las funciones que devuelven varios valores, como las funciones ligadas a las bases de datos, lo hacen en forma de tabla. En PHP los arreglos no son uniformes, es decir, pueden contener diferentes valores de diferentes tipos.

Función array: Crear una matriz

Formato: variable = array (valores)

Devuelve una matriz con los parámetros que se le pasan. A dichos parámetros se les puede dar un índice usando el operador =>

Nota: array() es una construcción del lenguaje que se utiliza para representar matrices literales, no una función regular.

La forma "índice => valor" separada por comas, define índices y valores. el índice puede ser de tipo cadena o numérico. Cuando el índice es omitido, se genera automáticamente un índice numérico, empezando en cero. Sí el índice es un entero, el siguiente índice generado será igual al índice con número mayor + 1. Note que cuando se definen dos índices idénticos, el último sobre escribe al primero.

Tener una coma después del último elemento de la matriz, aunque inusual, es sintácticamente válido.

A continuación crearemos unos ejemplos que hacen uso de arreglos:

Ejemplo arreglo1.php

```
<?php
```

```
$ciudad[] = "Coro";  
$ciudad[] = "Maracaibo";  
$ciudad[] = "Caracas";  
$ciudad[] = "Valencia";  
$ciudad[] = "San Cristóbal";
```

```
echo "Soy de $ciudad[3]";
```

```
?>
```



CURSOS Y TALLERES

XHTML1.0 & CSS

GNU/Linux Básico

PHP y MySQL

CANAIMA Educativo

OpenOffice / LibreOffice

Interesados dirigirse a:
Unidad de Computación.
Preguntar por: Lic. Luisa Ochoa



Dirección: Av. Universidad - Naguanagua

Observe que no se ha colocado ningún sub-índice, PHP lo va agregando a medida que se genera, hay que tener en cuenta que siempre el primer índice es 0 a menos que se indique lo contrario.

Otra forma de hacer un arreglo y que se vería mucho mejor es la siguiente:

Ejemplo arreglo2.php

<?php

```
$ciudad = array("Coro",
"Maracaibo", "Caracas",
"Valencia", "San Cristóbal");
echo "Soy de $ciudad[3]";
```

?>

Sí deseamos que el primer índice sea el 1, debemos hacer lo siguiente:

Ejemplo arreglo3.php

<?php

```
$ciudad = array(1=>"Coro",
"Maracaibo", "Caracas",
"Valencia", "San Cristobal");
echo "Soy de $ciudad[3]";
```

?>

Por lo que al usar el anterior ejemplo se imprimiría Caracas en lugar de Valencia.

Como hemos visto una matriz es el conjunto de variables con el mismo nombre pero con diferentes índices, a cada integrante de la matriz se conoce como elemento, los índices normalmente son números enteros pero se puede emplear caracteres, solo habrá que colocarlo entre comillas dobles ("), este tipo de uso se denomina matriz asociativas.

Ejemplo:

```
$capital["CO"] = "Valencia";
$capital["DC"] = "Caracas";
```

Podemos crear arreglos bidimensionales, como por

ejemplo:

Ejemplo arreglo_bi.php

<?php

```
$variable [0][1] = 1;
$variable [0][2] = 2;
$variable [1][1] = 1;
$variable [1][2] = 2;
```

```
echo $variable [1][1];
```

// Imprime 1

?>



Usando Array para crear una matriz bidimensional:

Ejemplo array_bi.php

<?php

```
$fruta = array( "pera" =>
array("color" => "amarillo",
"sabor" => "dulce",
"naranja" => array("color" =>
"naranja",
"sabor" => "dulce",
"limon" => array("color" =>
"verde",
"sabor" => "acido")));
```

```
echo $fruta["limon"]["sabor"];
echo $fruta["naranja"]["color"];
```

?>

Estructuras de Control

Ejecución condicional (IF)

Como todo lenguaje el PHP posee una de las más importantes características de muchos lenguajes. Permite la ejecución condicional de fragmentos de código. PHP caracteriza una estructura if que es similar a la de C:

El formato sería:

```
if (expr)
{ sentencia(s);
o bloques de sentencias;
}
```

expr se evalúa a su valor condicional (boolean). Si expr se evalúa como TRUE (verdadero), PHP ejecutará la sentencia, y si se evalúa como FALSE (falso) - la ignorará.

Las sentencias if se pueden anidar indefinidamente dentro de otras sentencias if, lo cual proporciona una flexibilidad completa para ejecuciones condicionales en las diferentes partes de tu programa.

A menudo queremos ejecutar una sentencia si se cumple una cierta condición, y una sentencia distinta si la condición no se cumple. Esto es para lo que sirve else.

```
if (expr)
{ sentencia(s);
o bloques de
sentencias; }
```

```
else
{ sentencia(s);
  o bloques de sentencias; }
```

La sentencia else se ejecuta solamente si la expresión if se evalúa como FALSE.

Empero podemos ir más allá en el condicionamiento de la ejecución de código usando elseif, como su nombre sugiere, es una combinación de if y else. Como else, extiende una sentencia if para ejecutar una sentencia diferente en caso de que la expresión if

original se evalúa como FALSE. No obstante, a diferencia de else, ejecutará esa expresión alternativa solamente si la expresión condicional elseif se evalúa como TRUE.

```
if (expr)
{ sentencia(s);
  o bloques de sentencias; }
elseif (expr) { sentencia(s);
  o bloques de sentencias; };
else { sentencia(s);
  o bloques de sentencias; }
```

Hay que tener especial cuidado cuando comparemos, si por ejemplo queremos comparar el valor \$a con el \$b debemos usar if (\$a == \$b) porque si usamos if (\$a = \$b) estaremos asignando en lugar de comparar.

Existe una segunda versión del de igualdad, este operador usa tres signos de igualdad y evalúa a verdadero sólo si los valores son iguales y del mismo tipo de datos.

```
if ($numero ===
$numero_aleatorio)
```

Tenga en cuenta que puede usar el operador Not (!) de la siguiente forma:

```
if !($mes == "marzo) sí la
expresión es TRUE la cambiará
a FALSE, tome en cuenta que
esta sentencia no es lo mismo
que if ($mes != "marzo").
```

Switch

switch es similar a una serie de

sentencias IF anidadas en la misma expresión. En muchas ocasiones, se quiere comparar la misma variable (o expresión) con muchos valores diferentes, y ejecutar una parte de código distinta dependiendo de a qué valor es igual. Para ello sirve la sentencia switch.

Ejemplo switch.php

```
<?php
$i = 2;
switch ($i) {
  case 0:
    echo 'i igual 0';
    break;
  case 1:
    echo 'i igual 1';
    break;
  case 2:
    echo 'i igual 2';
    break;
}
```

AWVEN
 Twitter: @awvene
 Identi.ca: @awven
<http://www.awven.com>



TIPS

#Thumbnails de tus archivos de LibreOffice

Abrir consola

Escribir

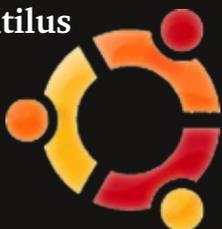
```
sudo add-apt-repository
ppa:flimm/ooo-
thumbnailer
```

```
sudo apt-get update
```

```
sudo apt-get install ooo-
thumbnailer
```

```
sudo apt-get install libgsf-
bin
```

```
killall nautilus
```



CUANDO ERAMOS LIBRES...



Cuando eramos libres y ni estábamos enterados

Fue entre los años 1997 y 1998 cuando comencé a trabajar por el software libre, en aquellos tiempos todavía el termino hacktivista no estaba en boga (al menos por estos lares), por ello no lo acuño. Recuerdo muy claramente que mi primera participación fue una breve charla en la facultad de ingeniería de la Universidad de Carabobo y la segunda un Festival de instalación junto con un Demo Day en un centro comercial en la avenida Bolívar de la ciudad de Valencia después no lleve mas cuentas.

En esos primeros años pude instalar Slackware, SuSE, Caldera y RedHat realmente no recuerdo cuantas distribuciones instale en mi equipo hasta poder dar con la que me sentí a gusto. Sin importar cual fuese la distribución seguía los pasos: a) Leer todo sobre el proceso de instalación, b) buscar información sobre la distribución y su configuración de acuerdo a los periféricos y/o



dispositivos con los que contaba, c) proceder a instalar, d) sí algo no funcionaba como debería, buscar en la red antecedentes y repararlo; y como paso final sí era necesario, el nefasto punto e) escribir a las listas de correo en búsqueda de ayuda. Y es que llegar al paso "e" podía llegar a ser lo mas difícil de la instalación de Linux, ya que no era suficiente elaborar un pregunta de forma inteligente sino aguantar los chaparrones de los mas versados y hasta de los neófitos que lo hacían solo por agregar unos correos mas al hilo de conversación. En estos días a los que pertenece esta parte de la historia solo existía en Venezuela Linux Preview de Francisco Bursi y

VELUG (Grupo de usuarios linux de Venezuela) todo lo demás surgio después. En estos días se aprendía mucho, para hacer una actividad lo único que hacia falta era tener ganas de hacerla y enviar un correo a la lista; y para el día, hora y lugar que fuera fijada la actividad solitos llegaban conversadores (speakers), curiosos y publico general; se hacían tres o cuatro instalaciones, se conversaba en torno al tema prefijado en la lista y cual futbolistas al final

de la jornada no faltaba el tradicional intercambio de CD's. No hacian falta presupuesto ni grandes auditorios, bastaba con una buena aula de clases y cada quien se trasladaba con sus propios recursos.

El crecimiento de la masa crítica, crítica.

Con el auge alcanzado por el software libre y por ende con Linux, se realizaron dos exitosos LinuxWeek, tras esto el ruido aumento y aparecieron mas "eventos" desplazando poco a poco a las pequeñas actividades, fue tal el éxito alcanzado por el movimiento que hasta el oído de políticos, ministros, periodistas y hasta especialistas en la materia sintieron su zarpazo; y fue entonces así que hasta se promulgo un decreto que "obligaba" el uso de software libre por parte de las instituciones del estado y muchos años mas tarde debido al pobre acogimiento de este apareció la resolución Nro. 025 que instruyo sobre el uso de Canaima GNU/Linux en estaciones de trabajo de órganos y entes de la Administración Pública y a esta le siguieron otras resoluciones que daban mas carácter a todo un marco para la justificación



VeneHosting.com

Tu Hosting en Venezuela

CUANDO ERAMOS LIBRES...

del uso del software libre, estas fueron la Resolución 026 que establece "los lineamientos de accesibilidad que deben ser aplicados en el desarrollo, implementación y puesta en producción de los Portales de Internet, a los fines de garantizar la accesibilidad de los ciudadanos y ciudadanas de modo rápido, oportuno y seguro a los recursos y servicios que ofrece el Estado Venezolano; y la Resolución 027 que establece "los requisitos mínimos para la prestación de servicios, por parte de personas naturales, en calidad de Facilitadores Comunitarios en el área de Tecnologías de Información Libres. En fin para este momento es de pensar que el software libre y Linux están consolidados en Venezuela y que todo aquel que tenga que decir algo en esta materia, puede hacerlo sin restricción alguna.

Hoy no puedo, mañana ?



Hoy día cuando para instalar Linux, perdón quise decir GNU/Linux, solo hace faltar poner

el CD o DVD en la ranura y dar cuatro clics; que tenemos un decreto y hasta lineamiento que "Obliga" el uso y promoción del software libre en Venezuela.

No se organiza una actividad porque no se cuenta con el

apoyo económico del sector publico o privado y hasta el "Guapo" del liceo me cobra 5000 por usar un aula.

No puedo ir a dar una charla a tal o cual lugar ya que no consigo quien me patrocine el traslado, hotel y comida.

No instalo XXXXX/Linux, es muy chimba, no instalo Canaima porque es socialista yo solo instalo y recomiendo usar Coolinux porque lo tiene todo, nadie me va a chalequear por usarlo, no hay que aprender nada y lo mas importante aun, es la mas ampliamente aceptada en el mundo entero, para todo lo demás existe Master Card. No programo con ningún lenguaje que se escriba con "P", los odio a todos, no los conozco, son ocho los monos.

No instalo ni ayudo, ni soy amigo de: Canaimistas, Ubunteros, Slackweristas, Debianistas, RedHaseros, Minteros y ???????? (reemplace los signos de ? por el gentilicio de su distribución favorita). Me quedo en casa.

Volvamos a ser

Quieres usar Linux y enseñar a otros a usarlo? olvídate de todo lo que leíste dos minutos atrás y... Con tus amigos de (el barrio, la urbanización, la cuadra, el colegio o la universidad) dispon de la sala de tu casa o un aula de tu escuela, elijan el tema de



conversación e instalen dos o tres distribuciones sin importar cuales sean.

NO obligues a nadie a usar la distribución que tú usas, invítalo a usar Linux y Software Libre, tú mismo descubre que puedes usar cualquier Linux y luego determinar cual es el que mas se adapta a tus necesidades. Ayuda a otros a encontrar la solución a sus problemas, no lo resuelvas tú solo ya que de lo contrario no estarás enseñando NADA.

Participa cuando quieras y/o puedas y mas aun, cuando te inviten; sin esperar patrocinios, hoteles 5 estrellas o comidas gourmet. La experiencia de participar te ayuda a crecer.

Enseña bien sobre lo que mejor haces, siempre hay alguien que estará agradecido

Comparte con tus amigos mas allá de la actividad del conocimiento, no solo de Linux vive el hombre... y las mujeres también.

No permitas que el ruido en la red (listas, blogs y redes sociales) te impidan hacer lo que te guste y para ti, sea lo correcto.

Alvaro Hernandez
Twitter: @ajha63

ANDROID (APK)

DESCARGANDO APLICACIONES ANDROID (APK) DESDE GNU/LINUX SIN USAR ANDROID MARKET



Alguna vez se han preguntado: ¿Cómo hago para tener mis programas favoritos en Android (.apk) sin necesidad de descargarlos del Market?.

Normalmente, siempre

estoy probando nuevas ROMS en mi teléfono/tableta y debo instalar a cada momento todo. Ustedes se preguntarán: si existe Titanium Backup u otros ¿Por qué no lo usa?. Lamentablemente esas herramientas no te dan el .apk. Adicionalmente toca hacer el proceso manual desde el Market: Imagínate que en un día instalas 10 ROMS diferentes, y a cada una le tienes que descargar el ClockWorkMod, dónde tienes que obligatoriamente autenticar tu dispositivo con Google para descargar 2.2 Mb en cada session, donde consigues consumirte 220 Mb de tu conexión 3G (que van desde 1 GB a 3 GB, dependiendo del plan contratado con tu operadora), ¿No es más sencillo crearse un repositorio local en la memoria SD del dispositivo y así tener disponible las aplicaciones cuando las requieras?.

El presente artículo tiene como finalidad mostrar una forma alternativa de descargar aplicaciones para Android (.apk) sin necesidad

de usar el Android Market.

¿Qué necesitamos?

1. Una distribución de GNU/Linux (en mi caso usé Linux Mint Debian Edition).
2. Tener instalado Android SDK[0].
3. Tener instalado PHP (aptitude install php5-cli php5-curl).
4. Tener instalado Wireshark (aptitude install wireshark).
5. Tener el teléfono/tableta rooteado.
6. Tener una cuenta Gmail (login/password).

Comenzamos

Lo primero que debemos obtener son los siguientes datos del dispositivo:

- userId: El id del usuario asociado al Market.
- deviceId: El id de nuestro dispositivo asociado al Market.

La única forma de conseguir estos datos es aplicando una técnica de sniffing[1] al teléfono. Como no existe una forma de colocar el USB en modo Trunk, entonces nos toca empotrar un sniffer (en mi caso elegí el tcpdump) a nuestro dispositivo:

TCPdump es un herramienta en línea de comandos cuya utilidad principal es analizar el tráfico que circula por la red. Permite al usuario capturar y mostrar a tiempo real los paquetes transmitidos y recibidos en la red a la cual el dispositivo está conectado.

TIPS

Instalar el menu clasico (testing) en Ubuntu 11.10
Usas Unity pero te gusta los clasico usa estos comando:

```
sudo add-apt-repository ppa:diesch/testing
sudo apt-get update
sudo apt-get install classicmenu-indicator
```

ANDROID (APK)

1. Nos descargamos el sniffer

```
# wget http://dl.0x29.com.ve/android/tcpdump
```

2. Montamos levantamos el servicio del ADB para comenzar la fiesta

```
# adb start-server
```

3. Remontamos el sistema de archivo en lectura-escritura

```
# adb shell mount -o remount,rw /system
```

4. Subimos el sniffer a nuestro sistema

```
# adb push tcpdump /system/xbin/
```

5. Le damos permiso de ejecución

```
# adb shell chmod +x /system/xbin/tcpdump
```

6. Comenzamos la captura de tráfico

```
# adb shell tcpdump -vv -s 0 -w /tmp/android.cap
```

Para ello debemos abrir el Android Market y instalar cualquier aplicación, matamos el proceso con CTRL+C y nos descargamos el dump generado por nuestro sniffer.

```
# adb pull /sdcard/android.cap
```

El Análisis

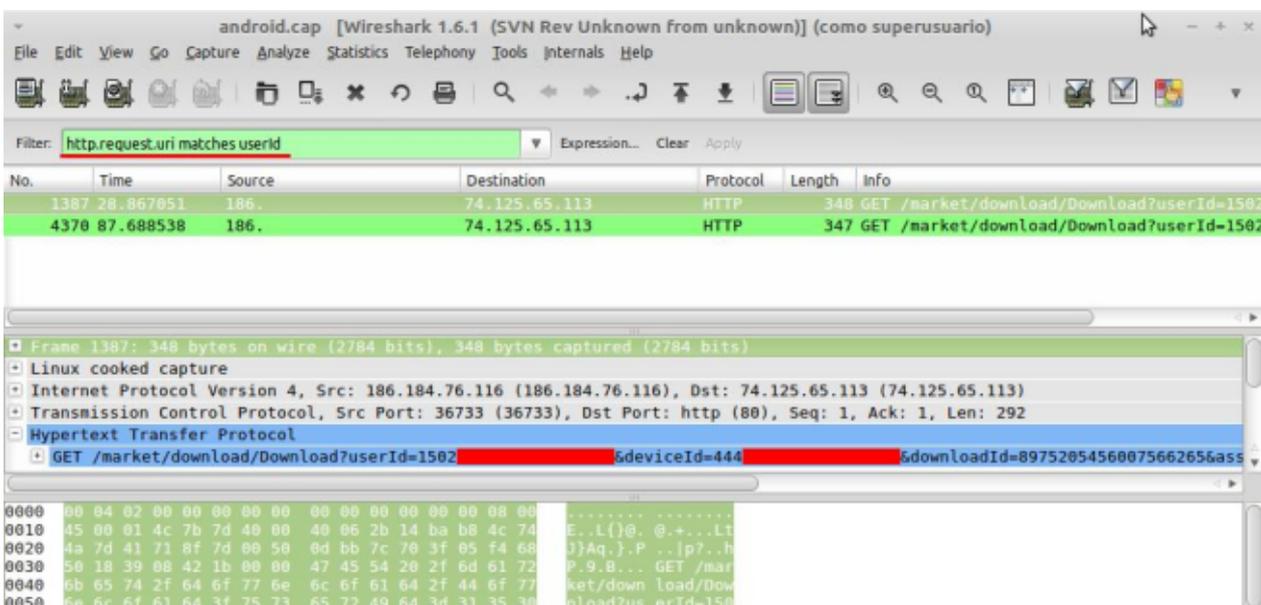
Al tener la muestra generada por el tcpdump simplemente procedemos a analizarla con Wireshark. El datos que nos interesa es HTTP Requests GET, que podremos obtenerlo desde el mismo tcpdump usando de argumento "port 80 and tcp[((tcp[12:1] & 0xf0) >> 2):4] = 0x47455420" ó simplemente abriendo la captura con el Wireshark y colocándole en la parte de "Filter" lo siguiente: http.request.uri matches userId.

Notamos una salida de la siguiente forma:

```
1387 28.867051 186.184.XX.YYY
74.125.65.113 HTTP 348 GET
/market/download/Download?userId=YYYYYY
YYYYYYYYYYYYYYYY&deviceId=ZZZZZZZZZZZZZZ
ZZZZZZ&downloadId=8975205456007566265
&assetId=-1220012608108377957 HTTP/1.1
```

Con eso obtendremos el userId y deviceId de nuestro teléfono.

Wireshark es un analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones, para desarrollo de software y protocolos, y como una herramienta didáctica para educación.



ANDROID (APK)

La carpintería

Solo nos queda descargarnos el Android Market API escrito en PHP para poder descargarnos[2] los .APK y el script que nos permitirá conectarnos al Market. Debemos descomprimir y desempaquetar el archivo y descargar el archivo dw_apk.php en el directorio examples:

```
# wget
http://dl.0x29.com.ve/android/dw_apk.phps -o
dw_apk.php
```

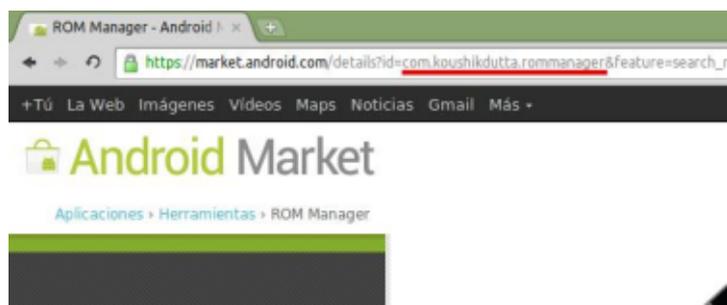
Hay que setear las siguientes variables con los datos extraídos en el archivo dw_apk.php:

```
$GOOGLE_EMAIL="TUCUENTA@gmail.com";
$GOOGLE_PASSWD="TUPASSWORD";
$ANDROID_FAKEDeviceID="01234567891234
56";
$ANDROID_DEVICEID="ELDATOQUESACASTED
ELDUMPLLAMADODEVICEID";
$ANDROID_USERID="ELDATOQUESACASTEDE
LDUMPLLAMADOUSERID";
```

Ahora vamos al Android Web Market[3], buscamos una aplicación GRATUITA (no sirve para descargar aplicaciones pagas) y extraemos de la Query String el nombre (por ejemplo, details?id=com.koushikdutta.rommanager).

Y ejecutamos

```
# php dw_apk.php
com.koushikdutta.rommanager
```



```
gon examples # php dw_apk.php com.koushikdutta.rommanager
* About to connect() to android.clients.google.com port 80 (#0)
* Trying 72.14.204.100... * connected
* Connected to android.clients.google.com (72.14.204.100) port 80 (#0)
> POST /market/api/ApiRequest HTTP/1.1
User-Agent: User-Agent: AndroidDownloadManager
Host: android.clients.google.com
Accept: */*
Cookie: ANDROIDID=DQAAAMAAAAA=HUAdxSP5pds6I2CRR8tYlfm7do6y-5Aky-I-3HzsDDUk
g75SxAn4y3H-vhFnR-tfEF3D75FwbQIJ4PCzWuCVcPPRpy-uLx3eYbfkcy6XQzWDoibeeYx4
g3b9xtV5FejautcorfxqGcmvvnCLJ3w
Content-Type: application/x-www-form-urlencoded
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Content-Length: 546

< HTTP/1.1 200 OK
< Content-Type: application/binary
< Date: Mon, 21 Nov 2011 14:57:31 GMT
< Expires: Mon, 21 Nov 2011 14:57:31 GMT
< Cache-Control: private, max-age=0
< X-Content-Type-Options: nosniff
< X-Frame-Options: SAMEORIGIN
< X-XSS-Protection: 1; mode=block
< Content-Length: 168
< Server: GSE
*
* Connection #0 to host android.clients.google.com left intact
* Closing connection #0

Downloading ROM Manager (-8914111577147149262)
```

Obtenemos la siguiente salida:

Y verificamos con un ls si la aplicación se

```
gon examples # ls -alh *.apk
-rw-r--r-- 1 root root 2,4M nov 21 10:29 com.koushikdutta.rommanager.apk
```

descargó correctamente:

Referencias

- [0] <http://developer.android.com>
- [1] http://es.wikipedia.org/wiki/Analizador_de_paquetes
- [2] <https://github.com/splitfeed/android-market-api-php/zipball/master>
- [3] <https://market.android.com>

Juan Angulo Moreno
Twitter: @apostols



gtkOrphan chao huérfanos

GtkOrphan: Elimina los huérfanos!

GtkOrphan es una aplicación gráfica para los sistemas Debian o basados en ella (Ubuntu, LinuxMint por ejemplo) que soporten Perl/Gtk2 para analizar el estado de sus instalaciones, en busca de bibliotecas huérfanas.

Esta aplicación está en los repositorios unstable/testing

Es bastante sencillo de usar y una forma eficaz de mantener nuestro sistema limpio, ahorrando espacio, ya que eliminamos librerías que no cumplen ninguna función o sencillamente están solo ocupando espacio.

Versión actual:
gtkorphan_0.4.4-1_all.deb

Las dependencias necesarias son:

gksu,
perl,
deborphan(>=1.7.17)
libgtk2-perl (>=1.100-1),
libglib-perl (>= 1.100-1),
liblocale-gettext-perl,
libgtk2-gladexml-perl

Para instalarlo desde consola:

```
sudo apt-get install gtkorphan
```

si tienes aptitude instalado

```
sudo aptitude install gtkorphan
```

y correrlo

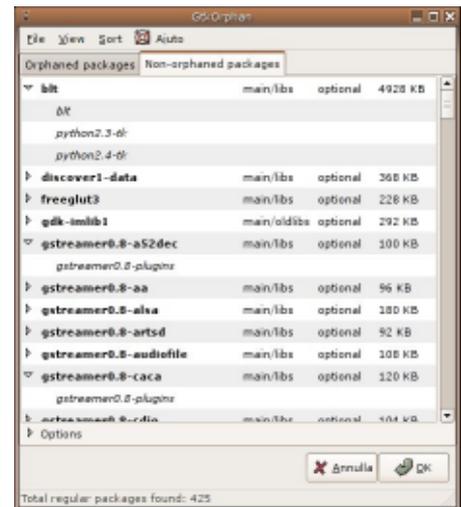
escribiendo en consola:

```
sudo gtkorphan
```

Nota: Es recomendado para usuarios avanzados.

Sitio Oficial

<http://www.marzocca.net/linux/gtkorphan.html>



TIPS

#Actualizar al Kernel 3.1.4 en Ubuntu / LinuxMint

Ingresar a la siguiente dirección:

<http://kernel.ubuntu.com/~kernel-ppa/mainline/v3.1.4-oneiric/>

Y descargar e instalar en este orden:

http://kernel.ubuntu.com/~kernel-ppa/mainline/v3.1.4-oneiric/linux-headers-3.1.4-030104_3.1.4-030104.201111281851_all.deb

http://kernel.ubuntu.com/~kernel-ppa/mainline/v3.1.4-oneiric/linux-headers-3.1.4-030104-generic_3.1.4-030104.201111281851_i386.deb

http://kernel.ubuntu.com/~kernel-ppa/mainline/v3.1.4-oneiric/linux-image-3.1.4-030104-generic_3.1.4-030104.201111281851_i386.deb

Si has instalado la versión de 64 descarga amd64.deb pero el linux-headers-3.1.4-030104_3.1.4-030104.201111281851_all.deb es para todos

Luego

```
sudo update-grub
```

Reinicia y listo!

LDAP y aplicaciones web

Tendiendo puentes entre servicios de directorio basados en LDAP y aplicaciones Web

La venerable [1] familia de protocolos X.500 de la ITU, y en específico el protocolo LDAP[2] derivado de DAP, buscan representar la información de directorio, concebido como un conjunto organizado de registros, utilizando operaciones y estructuras normalizadas.

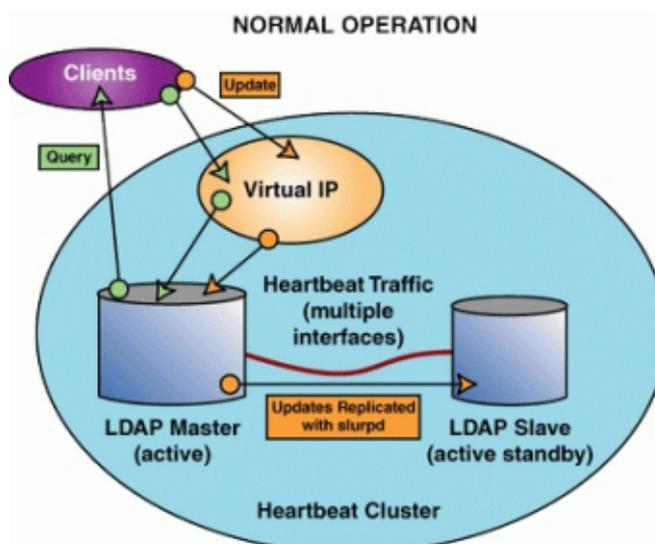
Su arquitectura, diseño, desarrollo, implementación y mantenimiento ha sido tradicionalmente llevada a cabo por los administradores de sistemas responsables de la infraestructura de TI, pero sus beneficios se han extendido hasta aplicaciones Web, dominadas por los responsables de desarrollo.

Por ello, la adopción de LDAP se ha visto severamente limitada dada la brecha de interpretación tecnológica que tradicionalmente ha existido entre desarrolladores y administradores, sumando a esto la desafortunada suerte que corren tecnologías de valor, incluso aunque sean estándares abiertos, como IPv6 o, en este caso, LDAP.

En este artículo estudiaremos alternativas para el cierre de esta brecha y el uso efectivo de LDAP en entornos donde las aplicaciones de línea de negocio están orientadas a la

Web.

Se sabe que ni un estándar ni un protocolo garantizan o representan una oportunidad tangible para el despliegue y aprovechamiento de una plataforma tecnológica, así que la fortaleza de LDAP en la actualidad radica en sus implementaciones, de las cuales las más populares son Microsoft Active Directory, IBM Tivoli Directory Server, Novell eDirectory y OpenLDAP [3], objeto de este artículo.



OpenLDAP es la implementación de código abierto más popular, cuya característica técnica más atractiva es que permite utilizar distintos repositorios para la data. El más popular, back-bdb, utiliza una base de datos incrustada de alto rendimiento para almacenamiento de estructuras clave/valor llamada BerkeleyDB, también de código abierto como parte de su licenciamiento dual, y organiza la data en árboles B+.

A diferencia de una base de datos relacional, como PostgreSQL o SQLite, la organización en árboles B+ es consistente con la información jerárquica nativa de un directorio organizacional. De allí que OpenLDAP implementa las operaciones del protocolo LDAP sobre el uso de BerkeleyDB con árboles B+ (existen, por supuesto, otros motores) y adiciona características como control de acceso, replicación de uno y

varios maestros, cifrado, capas (overlays) y distintos módulos para operaciones comunes como cambios de contraseña.

En la mayoría de los casos, OpenLDAP se utiliza para

almacenar la información de usuarios, grupos y recursos de una organización, incluyendo atributos personales, de permisologías, contraseñas, y muchos otros. Estos atributos vienen definidos en clases de objeto que vienen con el software, se consiguen en el dominio público o se escriben de acuerdo a las necesidades. Dentro de OpenLDAP, añadir más o menos atributos a los objetos de una clase es relativamente económico

LDAP y aplicaciones web

en comparación con una sentencia ALTER TABLE, la operación correspondiente en una base de datos relacional.

Así mismo, por su arquitectura, un repositorio basado en OpenLDAP es altamente escalable, y es fácil hacer pruebas con más de 1 millón de registros en computadores de sobremesa. Existen muchas implementaciones operativas de LDAP con decenas de millones de registros, muchas de ellas utilizando OpenLDAP. Así que el esquema es flexible y también escalable.

Visto desde un punto de vista administrativo, las ventajas del uso de LDAP, que han llevado a que renombrados profesionales de informática en la región lo apoden "LDAP: Lo Debes Aplicar Pronto", son:

1. Centralizar el manejo de identidades en la organización, obteniendo un control óptimo de la identidad de los usuarios de distintos sistemas
2. Gestionar las políticas de contraseña de una organización, incluyendo expiración, complejidad e históricos
3. Manejar registros centralizados de ingresos fallidos y exitosos, así como acceso a ciertos recursos
4. Facilitar el manejo de grupos, roles y permisologías para distintos sistemas
5. Almacenar de manera eficiente atributos comunes de los usuarios, desde

información de Recursos Humanos hasta datos de seguridad en diversos sistemas

No es posible cubrir todos los aspectos de una implementación adecuada de OpenLDAP en una organización en este artículo, por lo que recomendamos al lector consultar el "Manual de Referencia del Administrador de OpenLDAP", disponible en su página Web, y "LDAP System Administration", Carter, G., O'Reilly 2003. Así mismo resulta conveniente consultar bibliografía relacionada con servicios comunmente asociados con LDAP, como "Kerberos: The Definitive Guide", Garman, J., O'Reilly 2003 o "Using Samba", Carter et al., O'Reilly 2007.

Supongamos que un administrador de sistemas con experiencia en sistemas LDAP ha configurado OpenLDAP para almacenar la información de usuarios, grupos y contraseñas de una organización, y ha configurado sistemas operativos de cliente y servidor, de múltiples proveedores, y decenas de servicios desde appliances hasta bases de datos para interoperar con el protocolo LDAP y garantizar el uso de este repositorio centralizado.



El último eslabón de la cadena son las aplicaciones Web, que usualmente estarán fuera de su alcance y requerirán del consenso entre los responsables de infraestructura, los de bases de datos y los de diseño de software, e inclusive los de mantenimiento del ciclo de vida del software una vez que este entra en producción, si la organización así lo requiere.

La forma más sencilla de conectar una aplicación Web existente a un directorio LDAP es documentarse sobre las librerías o APIs [4] que, con casi toda seguridad, existen para conectar el lenguaje de programación utilizado con LDAP. Existen librerías de código abierto y gratuitas para virtualmente todos los lenguajes de programación de uso empresarial del mercado. Desde la más antigua, para el lenguaje C, hasta lenguajes tradicionalmente restringidos a la Academia, como Haskell. En el camino encontrará librerías para Java, PHP, Perl, Python, Ruby, C#, entre otros.

Para conectar la aplicación a OpenLDAP se requiere tener la librería y su documentación, y tener conocimiento de los conceptos de LDAP que son transversales a todas las implementaciones, como por ejemplo LDAP URIs,



bases de búsqueda, filtros, clases de objetos y atributos de objetos y operaciones. Todos estos conceptos están cubiertos en las referencias citadas en este artículo, pero, ¿es responsabilidad del programador de la aplicación conocer estos elementos? ¿merece la pena reescribir una aplicación solo para conectarla a LDAP?

Si está dudando en su respuesta a ambas preguntas, este artículo también explora una forma emergente de conectar aplicaciones Web a servicios LDAP como OpenLDAP, y es basada en la serialización de LDAP sobre JSON [5] para alimentar interfaces enriquecidas en aplicaciones Web, especialmente aquellas diseñadas bajo paradigmas AJAX.

Bajo este concepto, se puede programar una interfaz REST [6] que sirva como puerta de enlace entre el mundo LDAP y las aplicaciones Web, y utilice JSON como representación de la data. En este escenario, el programador solamente consulta los servicios Web previamente documentados y consume el JSON correspondiente. Inclusive se pueden utilizar marcos de trabajo como ExtJS, JQuery y

muchos otros, que interpretan este JSON y lo convierten en interfaces altamente interactivas para la Web, con poco trabajo del programador.

Además, este modelo puede resolver el dilema de gobernabilidad, ya que la puerta de enlace puede ser programada por los administradores con los requerimientos de los desarrolladores, y mantenida por los administradores beneficiando a los desarrolladores.

Para finalizar, y como prueba de concepto, utilizaremos un lenguaje de programación muy popular entre administradores de sistema para crear un servicio REST que se conecte a LDAP y serialice en JSON. Utilizaremos el marco de trabajo Dancer, que usted puede instalar en su sistema operativo Linux, OSX o Windows usando CPAN [7][8].

El contenido de la aplicación que proponemos como prueba de concepto es:

```
use Dancer;
use Net::LDAP;

set serializer => 'JSON';
any '/all' => sub {
    my %results; my $ldap =
    Net::LDAP-
```

```
>new('db.debian.org');
my $mesg = $ldap-
>search(base=>'dc=debian,dc=
org',filter=>'uid=bureado');
foreach my $entry ( $mesg-
>entries ) {
    undef %results; my %struct;
    foreach my $attr ( $entry-
>attributes ) {
        $struct{$attr} = $entry-
>get_value($attr, 'asref' => 1);
    }
    $results{$entry->dn} =
    \%struct;
}
return \%results; #
Devolvemos una estructura
arbitraria a JSON
};
```

Al inicializar esta pequeña aplicación en Dancer dispondremos de un método RESTful en la URI /all, que podremos llamar por GET o POST y nos arrojará una estructura JSON obtenida del directorio LDAP que podremos usar en nuestra aplicación Web; en este caso:

```
{"uid=bureado,ou=users,dc=de
bian,dc=org":{"shadowMax":["9
9999"],"uid":["bureado"
],"cn":["Jose"],"ircNick":["burea
do"],"uidNumber":["2895"],"obj
ectClass":["top",
"innetOrgPerson","debianAccou
nt","shadowAccount","debianD
eveloper"],"shadowMin
":["0"],
```

LDAP y aplicaciones web

```
"shadowWarning":["7"],"gidNumber":["800"],"jabberJID":["josparrella@gmail.com"],"labeledURI":["http://www.bureado.com/"],"gecos":["Jose Parrella,,,,"],"shadowLastChange":["14082"],"sn":["Parrella"],"keyFingerPrint":["7C1081B53C566C78AC7D3A2D51602C8D005C3B82"]}}
```

Bajo este concepto se pueden crear métodos determinísticos para el control de acceso en distintos módulos de nuestra aplicación, por ejemplo, al llamar /acceso/usuario/vista nuestra aplicación podría verificar en OpenLDAP si el usuario tiene acceso a la vista y devolver un objeto JSON con mensajes de autorización.

Al ser un servicio enteramente basado en HTTP, puede servirse utilizando Apache o cualquier otro servidor Web, como Nginx, para garantizar que solamente la aplicación pueda acceder a la puerta de enlace y pueden mantenerse los registros de acceso así como control de cortafuegos. Finalmente resaltamos que el objetivo de esta técnica es cerrar la brecha existente entre

administradores y desarrolladores y facilitar la adopción de tecnologías como LDAP en aplicaciones Web de forma sencilla y sustentable.

Nota al margen: desde la primera implementación exitosa de esta metodología en el primer trimestre de 2010 han surgido algunos productos, no necesariamente de código abierto, para facilitar estas integraciones (p.ej., Nimbus DS)

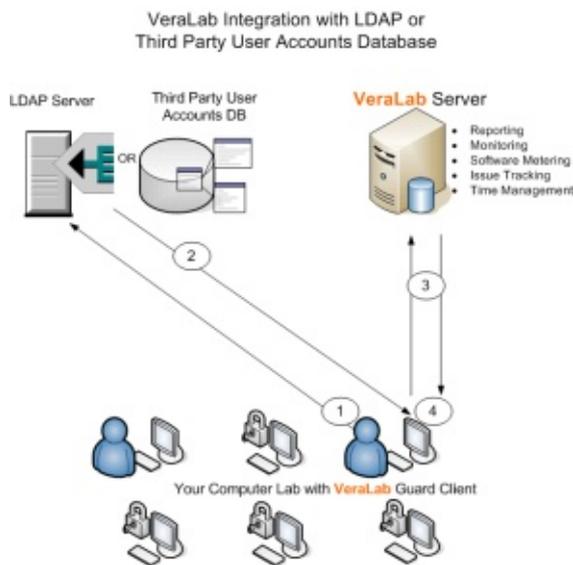
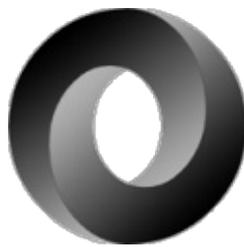
Notas:

- [1] Aprobados en 1988
- [2] Protocolo ligero de acceso a directorios, por sus siglas en inglés.
- [3] Los lectores experimentados notarán que,

OpenLDAP, y la adopción de ApacheDS es aun muy baja, incluso que la de productos propietarios de uso muy específico.

- [4] Interfaces de programación de la aplicación, por sus siglas en inglés.
- [5] Notación de objetos de JavaScript, por sus siglas en inglés.
- [6] Transferencia de estado representacional, por sus siglas en inglés.
- [7] <http://www.cpan.org/>
- [8] <http://perldancer.org/>

José Miguel Parrella Romero
Web: bureado.com
Twitter: [@bureado](https://twitter.com/bureado)



por razones de espacio, no he nombrado otros dos productos de código abierto que son muy populares, como Apache Directory Server y 389 Directory Server, antiguamente FDS; en el caso de FDS, está basado en

TIPS

Cacheando los DNS

Abrir consola

```
sudo apt-get install dnsmasq
```

Luego editamos resolv.conf

```
sudo gedit /etc/resolv.conf
```

Eliminamos el contenido y escribimos

```
nameserver 127.0.0.1  
nameserver 8.8.8.8
```

Ahora:

```
sudo /etc/init.d/networking  
restart
```

Ubuntu se la juega

Ubuntu se la juega en sus cambios.

Indudablemente que en este año que finaliza nuestros amigos de Canonical han captado la atención tanto de usuarios como de otras organizaciones o comunidades de Software Libre y Distribuciones. Sus apuestas a los cambios radicales, a muchos le ha parecido una osadía, otros un riesgo, a otros le parece una desfachatez, a otros una bendición, y a otros simplemente no les importa, en fin; esto simplemente es el color donde tantas personas con diferentes perspectivas comparten un sistema operativo que ha venido llevando por decirlo así la batuta en los últimos años,

Mucho más allá del simple hecho que esta distribución Linux este orientada al usuario final por su sencillez y solidez, Ubuntu no escapa de los usuarios que aman a Linux precisamente por su capacidad de configuración y adaptación, Linux es un sistema que ha crecido en comunidad, y que cada comunidad o individuo prueba, inventa, cambia hace y deshace de él. Es precisamente esta la imagen que a los usuarios finales ordinarios tienen de los Linuxeros, alguien metido en una pantalla negra haciendo y deshaciendo por horas, en un garaje o sótano de su casa; muchos



cables un desorden mezclado de tecnología y restos de comidas rápidas, entre otras cosas. Estos Linuxeros aman Linux por la oportunidad de personalizarlo, hacer y deshacer en él.

En el momento que Ubuntu hace estos cambios tan radicales orientándose un poco a las nuevas tecnologías, a los nuevos equipos y con una visión futurista diferente a lo ya acostumbrados, saltan las críticas y controversias al respecto, ya que las apuestas a Unity y a Gnome-Shell

representan reaprender cosas nuevas, menos oportunidad de configuraciones que ya estaban acostumbrados, y perder un poco eso de que linux funciona hasta en maquinas de bajo rendimiento. Si bien es cierto que estos cambios implican drásticamente una visión diferente de el uso de nuestras computadoras, y que Ubuntu a pesar de querer mantenerse orientado a un usuario final ordinario, es también siendo usado por Linuxeros expertos; quizá la única distribución con cambios a visión de futuro fuera de lo que en otras distribuciones ya nos tienen acostumbrados, es Ubuntu, tan es así, que otras distribuciones, han seguido estos cambios. Hay otras que

aun están renuentes entre ellas una basada en Ubuntu como lo es Linux Mint.

Linux Mint se mantiene aun con su entorno de escritorio más clásico y a pesar que entre Mint y Ubuntu no hay mayor diferencia, algunos prefieren usar Mint que Ubuntu precisamente por esos cambios, pero entonces nos cabe unas preguntas:

¿Es acaso el camino; mantenerse en un perfil para equipos de bajo rendimiento?

Esto implica no incluir aplicaciones que consuman muchos recursos, pero hoy en día las computadoras van creciendo en Hardware muy rápidamente, lo que implicaría pues ignorar entonces las capacidades de los nuevos usuarios que adquieren nuevas tecnologías y dirigir la visión a aquellos que tienen equipos en desuso o por estarlo. Esto sería entonces pues para muchos una decisión acertada o no.



LinuxMint, la distro ...



Linux Mint:
La distro del
usuario final

Antes de
comenzar ha expresar ideas
referentes a Linux Mint,
recurriremos a la Wikipedia.

Linux Mint es una distribución
del sistema operativo
GNU/Linux, basado en la
distribución Ubuntu (que a su
vez está basada en Debian).

Entre sus característica:
Modelo de desarrollo
FOSS

Última versión estable 11

Núcleo: Linux

Tipo de núcleo: Monolítico

Sistema de gestión de
paquetes: dpkg

Método de actualización: APT

Licencia: GPL

Estado actual: Estable

La rama inestable de Linux
Mint lleva por nombre código
"Romeo". No está activada por
defecto en los lanzamientos de
la distribución. Los usuarios
que deseen conseguir las
características "más
avanzadas" y deseen ayudar a
la distribución probando los
nuevos paquetes, pueden
agregar la rama "Romeo" a sus
fuentes de APT. "Romeo" no es
un rama en sí misma y no

sustituye a los otros depósitos.

Linux Mint no sigue un ciclo
predecible de lanzamientos
sino que los plantea uno tras
del otro. En cada caso, el
proyecto define primero los
objetivos del próximo
lanzamiento. Cuando se
alcanzan todos los objetivos,
usualmente, se procede al
lanzamiento de una Beta,
luego pasa por el lanzamiento
de una Release Candidate (RC)
y luego si todo va bien,
se anuncia la fecha
para el lanzamiento de
la versión estable.

La última versión
estable fue lanzada el
16/08/2011, lleva
como nombre Katya
conocida como
LinuxMint 11, basada
en Ubuntu 11.04

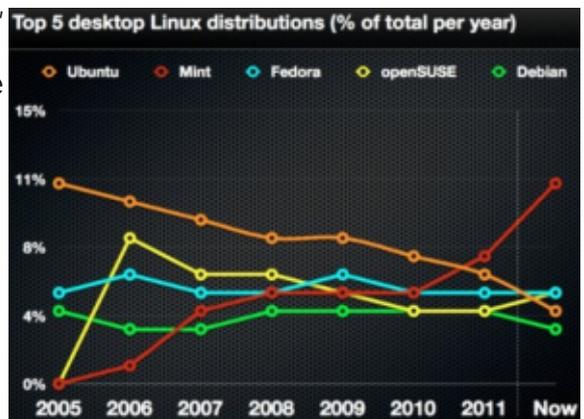
Natty, en dos ediciones
GNOME y LXDE.

Linux Mint siempre ha gozado
de una reputación muy bien
merecida por su excelente
compatibilidad, ya que ha
incluido una gran variedad de
codecs multimedia
propietarios que están
usualmente ausentes en las
grandes distribuciones.

La Principal Edición (main) de

Linux Mint proporciona un
entorno de escritorio con
GNOME y codecs multimedia,
todo estos contenidos en un
sólo CD. Está diseñada para
satisfacer a todos, usuarios
individuales (principiantes) y
profesionales. Una vez que lo
instalas ya tu máquina esta
operativa y configurada para
realizar diversas tareas.

El Lider del proyeco es
Clement Lefebvre, Jamie Boo



Birse es el mantenedor
edición KDE, Merlwiz79
mantenedor edición XFCE,
Kendall Weaver mantenedor
ediciones LXDE y FluxBox.

El día 09-11-2011 el famoso
sitio DistroWatch publicó que
Linux Mint había pasado al
primer lugar entre las
distribuciones, desplazando a
Ubuntu al segundo lugar.

En ese momento la tabla



VeneHosting.com

Tu Hosting en Venezuela

LinuxMint, la distro ...

ranking quedó:
Mint 2222
Ubuntu 2100
Fedora 1723
Debian 1318



Ahora, Linux Mint 12 "Lisa" RC basado en Ubuntu 11.10 Oneiric Ocelot, ha creado una controversia, al indicar que vendrá por defecto con escritorio Gnome3, pero con "MGSE" (Mint Gnome Shell Extensions) lo cual lo hace más amigable a los usuarios. MGSE hace que Gnome3 sea mucho más sencillo de usar, permite que se puedan deshabilitar componentes para obtener Gnome 3 puro, simplemente active o desactive lo que considere necesario.

También incluye MATE que es nada menos que un fork que hace que Gnome3 sea compatible con Gnome2. Gracias a MATE, se puede ejecutar las dos versiones de Gnome en el mismo sistema. Esta aplicación no es

completamente estable, pero hay una estrecha colaboración entre los desarrolladores de Linux Mint y MATE. Con el tiempo el proyecto ganará la madurez y ofrecer a los

usuarios una experiencia de escritorio tradicional y sólida.

Según los desarrolladores, MATE será en todos los sentidos idéntica a Gnome 2 y representa el

futuro de los escritorios tradicionales dentro de Linux; es visionario ese comentario esperaremos su evolución para poder opinar al respecto.

Los desarrolladores de LinuxMint presentan un entorno de escritorio más amigable, una mayor facilidad al momento de instalar programas, el uso de codecs privativos, lo que ha llevado que gane día a día más simpatizantes.

En el blog de LinuxMint se lee "Probablemente lanzaremos dos ediciones separadas, una para GNOME 2.32 y otra para GNOME 3.2. También estamos trabajando en colaboración con el proyecto MATE (que es un fork de GNOME 2) en estos momentos para ver si

podemos hacer compatibles estos escritorios en un esfuerzo que os

permitiría ejecutar tanto GNOME 2 (o MATE) como GNOME 3 en el mismo sistema, tanto en Linux Mint 12 como en futuras distribuciones."

Muchos de los usuarios "avanzados" de Linux que de una u otra forma han usado Mint han hecho buenos y malos comentarios sobre la nueva versión "Lisa".

Algunos dicen que Mint no ha inventado nada nuevo, como por ejemplo: las Gnome Shell Extensions es un concepto de 1995 que viene de NeXTSTEP (ver termino en wikipedia.org), pienso que si algo es bueno y se puede adaptar a lo que se tiene para mejorar debe hacerse.

En un post del sitio paraisolinux.com, hacen referencia a un texto publicado en el blog de LinuxMint y cito: "no pueden continuar usando Gnome2 pues es necesario abrazar nuevas tecnologías, pero que eso no significa que el usuario tenga que cambiar la forma en como usa su escritorio". Comparto ea idea, un +1 a los desarrolladores de LinuxMint.

Un número considerable de usuarios de Ubuntu con Unity, y de LinuxMint con Gnome-shell han empezado a pasarse a Xfce, incluso algunos de los usuarios de Fedora han empezado a usar este escritorio. Es lo bonito

LinuxMint, la distro ...

de Linux, podemos adaptarlo a nuestras necesidades y gustos.

Cabe destacar que Linus Torvalds, la considera su distro favorita.. por algo será.

Debemos tener en cuenta que una inmensa cantidad de usuarios abandonaron Ubuntu huyendo de Unity (Unity es un fork mal hecho de gnome shell), el cual considero más apropiado para tablets, Gnome es mejor en equipos de escritorio y laptops.

Igualmente en LinuxMint pueden usar diferentes escritorios tales como: XFCE, Gnome 2, Gnome 3 sin Shell, Gnome-Shell, KDE, LXDE, openBOX, FluxBox, y Unity.

Linus Torvalds llevo a sugerir que si no les gusta Gnome3 o Unity, pueden usar XFCE.

Linux Mint 12 "Lisa" RC como viene para su descarga funciona excelentemente y es bastante fácil de usar.

En un comentario de muylinux.com una persona que se llama Benitez escribe: "No cabe duda que supieron mezclar lo innovador con lo conservador" concuerdo con él/ella.

LinuxMint apuesta fuerte a GNOME 3, este año Linux Journal, lo declaró como Producto del año.

El día 24-11-2011 fue liberada



Linux Mint 12 "Lisa" en los próximos días estaremos observando el desenvolvimiento de esta versión.

Linux Mint 12 incluye:

- Kernel 3.0
- GNOME 3.2.1;
- MGSE (Mint Gnome Shell Extensions);
- MATE interfaz (fork de GNOME 2);
- Mozilla Firefox 8.0;
- Mozilla Thunderbird 7;
- Instalación de codecs multimedia y aplicaciones adicionales en un click;
- Nuevo tema de escritorio por defecto Mint-Z;
- Un nuevo conjunto de fondos de pantalla;
- LightDM;
- LibreOffice 3.4 suite de oficina.

En el día de su lanzamiento según distrowatch.com

1	Mint	5142
2	openSUSE	2933
3	Fedora	1728
4	Ubuntu	1698
5	Puppy	1506
6	Debian	1382

Ubuntu sigue en caída

vertiginosa, ojala la gente de Canonical en especial Mark Shuttleworth tome conciencia y "elimine" Unity.

LinuxMint, tiene un gran reto por delante, solo queda esperar si soportan la presión y continúan así... tal como van.

Recomiendo que uses en prueba GNOME Shell y por medio de la aplicación GNOME Tweak Tool puedes modificarlo y adaptarlo a tus gustos.

Larga Vida a LinuxMint, la distro del futuro!

Enlaces

Sitio Oficial:

<http://linuxmint.com>

Blog:

<http://blog.linuxmint.com>

Comunidad en Español:

<http://www.linuxmint-hispano.com/>

Twitter: @linux_mint

Héctor A. Mantellini
Coordinador Safecreative
Venezuela

<http://www.safecreative.org>

Coordinador Regional

AWVEN

<http://www.awven.com>

Twitter: @xombra

Identi.ca: xombra

<http://www.xombra.com>

GNU/Linux User:

#414452

Sitios Recomendados

<http://www.elandroidelibre.com/>

Sitio dedicado a Android en español, excelentes tips.



<http://elsoftwarelibre.wordpress.com/>

Sitio dedicado al software libre, publica las novedades de las principales distribuciones y aplicaciones.

<http://www.ubuntronics.com/>

Blog personal de Javier Ledesma, donde toca diferentes tópicos tales como: Android, Arch Linux, Fedora, GNU/Linux entre otros temas.



¿Qué es el Plan Ceibal?

El Plan Ceibal es un proyecto socio educativo implementado por el gobierno de la República Oriental del Uruguay. Creado por Decreto Presidencial con fecha del 18 de Abril de 2007, con el fin de proporcionar a cada escolar y maestro de la escuela pública una computadora portátil, capacitar a los docentes en el uso de dicha herramienta, y promover la elaboración de propuestas educativas acordes con las mismas. El Significado de la palabra "Ceibal" es: "Conectividad Educativa de Informática Básica para el Aprendizaje en Línea".

Objetivos del Plan Ceibal

El Plan Ceibal busca promover la inclusión tecnológica, con el fin de disminuir la brecha digital que existe con respecto a otros países. No, obstante la sola inclusión de la tecnología en las escuelas no asegura el cumplimiento de la meta si no se la acompaña de una propuesta educativa acorde a los nuevos requerimientos, tanto para maestros, sus alumnos y sus familias.

Es así que el Plan se basa en un completo sistema que busca garantizar el uso de los recursos tecnológicos, la formación docente, la elaboración de contenidos adecuados, además de la participación familiar y social.

Los principios estratégicos que encierra este proyecto son: la equidad, igualdad de oportunidades para todos los niños y todos los jóvenes, democratización del conocimiento, así como también de un aprendizaje, no solo a la educación que se les da en la Escuela, sino en aprender ellos mismos a utilizar una tecnología moderna.



El proyecto desarrolla una cultura colaborativa en cuatro líneas: niño-niño, niño-maestro, maestro-maestro y niño-familia-escuela. Promueve la veracidad y criticidad tecnológica en la comunidad pedagógica respetando a los principios éticos.

De igual modo, este sistema busca la formación y actualización de los docentes, así como

también la implicación y apropiación, tanto en el área técnica como en la pedagógica, facilitando el uso educativo de los nuevos recursos. Además genera sistemas de apoyo y asistencia técnico pedagógica específica destinada

a las experiencias escolares asegurando su adecuado desarrollo. De esta manera, involucra a los padres en el acompañamiento y promoción de un uso adecuado y responsable de la tecnología para el beneficio del niño y la familia.

El Plan Ceibal desde su incursión ha presentando avances significativos que a continuación se mencionan :

En mayo del 2007 se inicia una prueba piloto en Villa Cardal (departamento de Florida), con la puesta en marcha para 150 alumnos y sus profesores. Villa Cardal es un pueblo de 1.290 habitantes y una sola escuela de 150 niños. Para esta etapa se utilizan equipos que fueron donados por One Laptop Per Child. OLPC. A modo de plan piloto, este periodo sirvió para solucionar las complicaciones que todo gran proyecto encuentra al ponerse en marcha.

Para Agosto 2009 comenzó a ampliar gradualmente el alcance del Plan Ceibal a las instituciones de educación privadas. En octubre de ese mismo año se termina de completar el plan en todos los departamentos del interior del país.

En octubre del 2010 el Plan Ceibal comienza su segunda etapa; entregando computadoras a los alumnos del ciclo básico de enseñanza secundaria pública y a alumnos de UTU.

Componente de Hardware del Equipo

El aparato es pequeño, incluso demasiado para ser manejado por las manos de un adulto. El hardware de la máquina está diseñado para que permita una larga duración de la batería, no para ser extremadamente rápida. Las baterías tienen una duración de días, no de horas, gracias a un procesador con baja frecuencia de reloj.

El portátil posee dos grandes antenas de WiFi, que son al mismo tiempo los cierres de la tapa. No tiene disco duro sino memoria flash como dispositivo para almacenar el sistema operativo y los datos del usuario. La memoria flash puede expandirse por medio de unidades externas de tipo estándar, a través de sus tres puertos USB.

La tapa puede girarse totalmente y convertir el aparato en un tipo tableta sin teclado, aunque el siguiente prototipo XO-2 incluirá una pantalla táctil.

También llevan una webcam en la tapa, micrófono, dos altavoces, lector de tarjetas SD, varios botones tipo consola de juegos, y LEDs diversos para teclado y batería.

Componente de Software del Equipo

El sistema estaba basado inicialmente en una licencia GNU con núcleo Linux y un sistema de escritorio ultra simple en el que las ventanas siempre se encuentran maximizadas. Hay



controles alrededor de la ventana, en forma de marco, que pueden mostrarse u ocultarse mediante la presión de una tecla. La OLPC sólo puede realizar tareas básicas: escribir documentos, elaborar dibujos, entrar a Internet, juegos sencillos, escuchar música, ya que está diseñado para quienes nunca antes han tenido una PC.

Una de las piezas clave del proyecto en lo que se refiere al software de comunicaciones, consiste en que las unidades forman una red autogestionada, donde cada uno de los clientes es, al mismo tiempo, un enrutador. Así, la red extiende su cobertura gracias a la presencia de los propios aparatos, ya que cada uno es enrutador del siguiente, de manera que forman una cadena que no depende de nodos centrales.

La conectividad con otras máquinas está apoyada por un sistema de visualización del entorno local, cercano y lejano. Unas teclas de función ilustradas con símbolos sencillos acceden a estos tres niveles de visualización del entorno.

La otra pieza clave consiste en el empleo del famoso entorno educativo Squeak, que es un mundo de objetos interactivos con vida propia gracias al lenguaje Smalltalk (el propio Squeak está escrito en este lenguaje), mediante el cual niños de cualquier edad aprenden conceptos gracias a la experimentación directa con gráficas tortuga y multimedia.

Además de Squeak/eToys, el sistema contiene estas otras aplicaciones: navegador web, lector de RSS, videoconferencia,

un editor de texto derivado del Abiword, Tam-Tam (una aplicación sencilla de música) y Memory (un juego de memoria musical).

Usa como lenguajes de programación Python, JavaScript, Csound (lenguaje de síntesis sonora) y el propio entorno Squeak, aparte de los usados por otros programadores.

A continuación los requerimientos para optar y ser parte activa de este proyecto.

Pasos para realizar la compra de laptop del forma particular para alumnos de Colegios y Liceos Privados son los siguientes:

1. Todos los alumnos de Colegios Privados de 1° a 6° de Educación Primaria y Liceos Privados de 1° a 3° de Ciclo Básico podrán adquirir su laptop llevando adelante los pasos correspondientes para ello.

2. Los alumnos de Colegios Privados adheridos a Plan Ceibal podrán gozar de las bonificaciones existentes para la compra de laptops, en base a la anualidad que se pague.

3. Los alumnos de Colegios NO adheridos a Plan Ceibal, no accederán a ningún tipo de bonificación, abonando por el equipo el costo total correspondiente al mismo.

4. Para efectuar la compra será excluyente presentar copia del documento de identidad del usuario a quien se asignará la Laptop, junto con la Declaración Jurada que se completará a continuación. De no presentarse ambos documentos NO se efectuará la venta.

5. En caso de que quien firme la Declaración Jurada no pueda asistir en la fecha asignada para el pago y retiro del equipo, puede autorizar a otra persona a hacerlo, completando y entregando la siguiente CARTA, Carta de Autorización a Tercero.

6. Mas detalles en <http://latu30.latu.org.uy/>

Plan Ceibal y Premio Frida 2011.

El Plan Ceibal ha rendido sus frutos y ha sido galardonado con el Premio Frida, éste es el reconocimiento otorgado a los proyectos que más han contribuido al desarrollo de la sociedad de la información de América Latina y el Caribe.

Deseo manifestar un especial agradecimiento a todo el equipo que labora en el plan Ceibal, en especial a las personas que me suministraron la información necesaria para la elaboración de este artículo: Lic. Florencia González (Área de Comunicación y Realización Audiovisual), Lic. Alejandra Alcántara (Área de Comunicación y Realización Audiovisual), Inés Blixen (Portal Ceibal), Yeanina Merlo (Mesa de Ayuda), Natalia González (Mesa de Ayuda).

Para mayor información :
Plan Ceibal
www.ceibal.edu.uy

Autor:
Naudy Villarroel Urquiola
twitter: @naudyu

TIPS

#Instalar "Extra Fonts"

Abrir consola y escribir:
`sudo apt-get install ttf-mscorefonts-installer`

`cd /usr/share/fonts/truetype`

`sudo fc-cache -f -v i`

Reconstruir la información de las fuentes.

BleachBit el limpiador

BleachBit es una herramienta de código abierto para sistemas Linux para limpiar la información innecesaria de la computadora que ayuda a liberar espacio en disco.

Elimina:

- Caché,
- cookies,
- historial de Internet,
- archivos temporales, y desecha basura que no sabía que estaba ahí.



BleachBit incluye características avanzadas tales impedir la recuperación, limpiando el espacio libre en disco para ocultar los rastros de archivos borrados por otras aplicaciones.

- Borrar la memoria y swap en Linux
- Eliminar accesos directos rotos en Linux
- Eliminar el historial de Firefox sin eliminar todo el conjunto de archivos
- Eliminar idiomas que no se utilizan
- Eliminar la lista de documentos recientes OpenOffice y LibreOffice sin eliminar todo el archivo Common.xcu
- Sobrescribir el espacio libre en disco para ocultar archivos
- Limpiar bases de datos de Yum
- Reducir los archivos sin extraer los datos para ahorrar espacio y mejorar la velocidad

Para su descarga elige tu sistema operativo:

Sitio oficial

<http://bleachbit.sourceforge.net>



TIPS

#Reemplazar texto desde la terminal

Una manera sencilla para encontrar determinado texto y sustituirlo por otro, todo ello desde la consola; fácil y rápido.

Lo único que tienes que hacer es introducir en la consola lo siguiente:

```
find /home/jonateo/Desktop/ -name \*.txt -exec sed -i "s/casa/hogar/g" {} \;
```

Lo que hicimos fue: Sustituir todas las palabras "casa" por la palabra "hogar", de todos los archivos .txt que se encuentren en el escritorio.

visto en: <http://usemoslinux.blogspot.com>

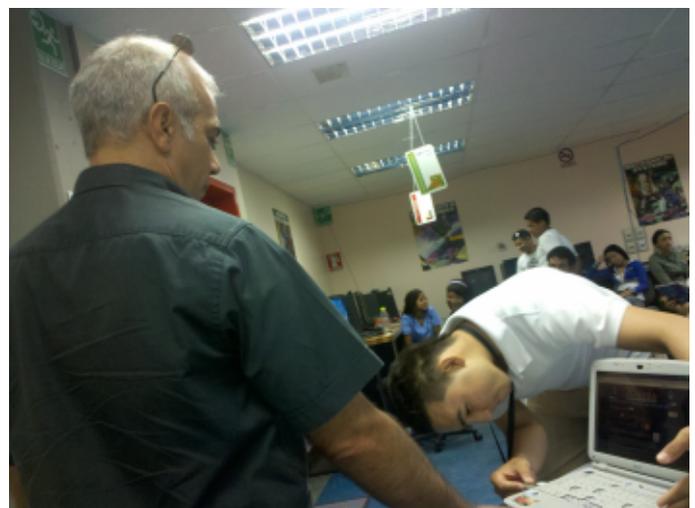
VaSlivre estuvo el día 04 de Noviembre en el INCES (Instituto Nacional de Capacitación y Educación Socialista) Los Colorados en Valencia - Carabobo. Fuimos invitados por la Lic. Liliana Vasquez para dictar dos charlas a los alumnos de informática sobre el lenguaje de programación PHP.

En esa oportunidad los ponentes fueron:

Angel Cruz (abr4xas) con la charla Que es PHP y **Héctor A. Mantellini (xombra)** con la charla Programación Segura en PHP.

Los alumnos mostraron gran interés en ambas charlas.

Luego de culminadas las charlas, los alumnos preguntaron por otros lenguajes de programación (Python, Ruby, Java y C++) y otras tecnologías involucradas, y del uso de Licencias de software.



PHPmonitor Monitoriza tu servidor(es)

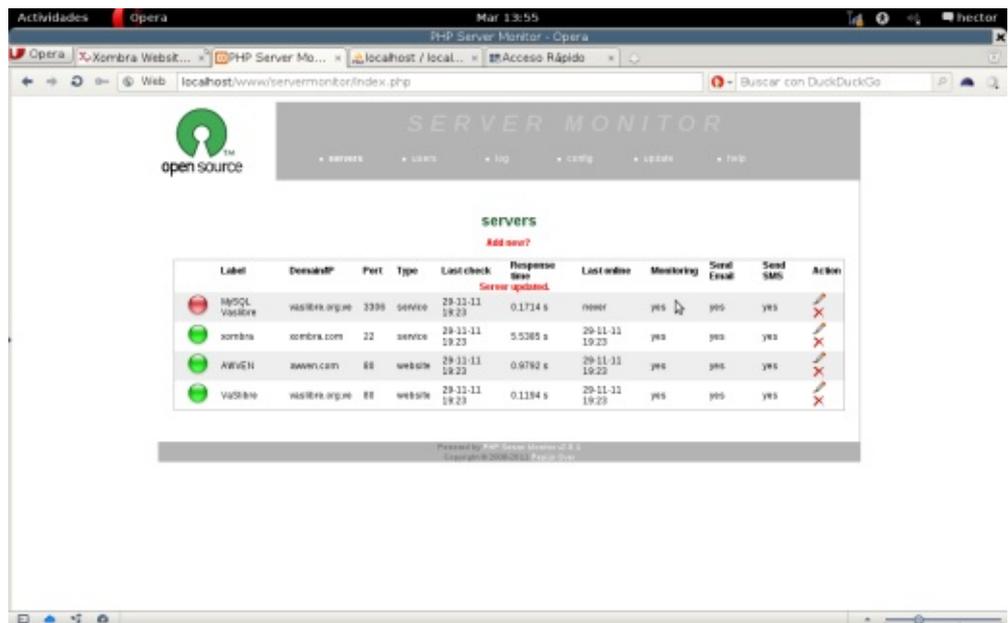
PHP Server Monitor es un script hecho en PHP que verifica que los servidores que tengas en una lista estén online y que los puertos y/o servicios estén en funcionamiento.

Viene con una interfaz de usuario basada en web donde puedes añadir y eliminar servidores o sitios web a partir de una base de datos en MySQL, además puedes administrar a los usuarios para cada servidor con un número de móvil y una dirección de correo electrónico.

En el "Agregar servidor" de la página, puede elegir si se trata de un "servicio" o un "sitio web":

- Un servicio de conexión se realizará a la ip o dominio entrante usando un puerto determinado. De esta manera se puede comprobar si ciertos servicios en el equipo se está ejecutando. Por ejemplo si quieres comprobar el servicio de IMAP por ejemplo, solo debes incorporar el puerto 143.

- Sitio web, usted puede introducir un enlace a un sitio web (por ejemplo <http://www.xombra.com/>), la



que se utilizará para abrir el cURL andcheck de la página web, verificando el estado HTTP. Si el código de ese estado HTTP está en el rango 4xx, significa que ha ocurrido un error y la página web no es accesible al público. En ese caso, el script devolverá una "línea de estado", y comenzará a enviar información ya sea por sms o a su correo electrónico.

Cómo utilizar:

1. Descargar el código desde: <http://sourceforge.net/project/s/phpservermon/>

- 2.- Extraer el contenido (phpservermon)

- 3.- Como requerimientos necesitamos:

1. php 5
2. cURL
3. MySQL
4. FTP

- 4.- Copiamos el archivo config.inc.php.sample como config.inc.php

- 5.- Editamos el archivo y cambiamos los valores

```
define('SM_DB_PREFIX', 'monitor_');
// Database username
define('SM_DB_USER', 'usuario');
// Database password
define('SM_DB_PASS', '112233445566');
// Database name
define('SM_DB_NAME', 'servermonitor');
// Database host
define('SM_DB_HOST', 'localhost');
```

- 6.- Crear bien sea por phpmyadmin u otra aplicación la base de datos "servermonitor" o el nombre que gustes

- 7.- Ejecutamos el archivo install

Ojo: cambia según tu ruta

<http://localhost/www/servermonitor/install.php>

renombramos o eliminamos el archivo install.php

8.- ahora:

<http://localhost/www/servermonitor/>

Podemos ir configurando según nuestro criterio

Nota: luego de agregar un nuevo servidor debes hacer click en "update"

Puedes añadir la actualización en un cron de esa forma podrás tener la información actualizada.

TIPS

#Instalar Gnome3 en Ubuntu / LinuxMint

"Es experimental no usar en entornos en producción"

Abrimos consola escribimos:

```
sudo add-apt-repository ppa:gnome3-team/gnome3
sudo add-apt-repository ppa:ubuntugnometeam/ppa-gen
sudo aptitude update
sudo aptitude dist-upgrade
sudo aptitude install ugr-desktop-g3 gnome-shell
sudo apt-get upgrade
```

Reinizamos y elegimos GNOME



Estructura de Carpeta

Estructura de Directorios y/o Carpetas en GNU/Linux

Los sistemas operativos respetables que siguen el estándar IEEE POSIX (Portable Operating System Interface)(El término fue sugerido por Richard Stallman en respuesta a la demanda de la IEEE, que buscaba un nombre fácil de recordar. Una traducción aproximada "Interfaz de sistema operativo portable" o "Interfaz portable de sistema operativo"). En la mayoría de distribuciones basadas en GNU/Linux se siguen estas recomendaciones, encontrando los siguientes directorios principales/:

/bin/: comandos para los usuarios del sistema.

/boot/: archivos necesarios para el arranque del sistema.

/dev/: dispositivos del sistema.

/etc/: archivos de configuración del sistema y de las aplicaciones que se encuentren instaladas

/home/: directorio para poner las carpetas home de los usuarios.

/lib/: librerías esenciales para el núcleo del sistema y módulos del mismo.

/media/: Contiene los puntos de montaje de los medios extraíbles de almacenamiento, tales como lectores de CD-ROM, Pendrives



(memoria USB), e incluso sirve para montar otras particiones del mismo disco duro, como por ejemplo, alguna partición que sea utilizada por otro sistema operativo.

/mnt/: punto de montaje temporal para dispositivos.

/opt/ Contiene Paquetes de programas opcionales de aplicaciones estáticas, es decir, que pueden ser compartidas entre los usuarios.

Dichas aplicaciones no guardan sus configuraciones en este directorio; de esta manera, cada usuario puede tener una configuración diferente de una misma aplicación, de manera que se comparte la aplicación pero no las configuraciones de los usuarios, las cuales se guardan en su respectivo directorio en /home.

/proc/: procesos y variables del kernel.

/root/: directorio del root del sistema.

/sbin/: comandos especiales que solo el root puede ejecutar.

/tmp/: donde se almacenan los archivos temporales. Según la distribución utilizada (o la configuración que utilizemos) se borran al arrancar el sistema o cada cierto período de tiempo.

/usr/: segunda estructura jerárquica, utilizada para almacenar las aplicaciones instaladas en el sistema.

/var/: directorio para los spoolers de impresión, ficheros de log, etc.

Importante:

Los directorios y/o carpetas creadas por el sistema operativo "NO DEBEN ELIMINARSE", debido a que podrían hacer al sistema inestable.

El Tao de la programación

El Tao de la Programación

En el mes de octubre leímos en el blog de José M. Aguilar (variablenotfound.com) un texto que llamo nuestra atención, "The Tao of Programming". libro escrito por en el año 1987 por Geoffrey James (geoffreyjames.com). Aún cuando lo expresado es en tono sarcástico el mensajes de fondo habla sobre calidad en la programación, gestión de proyectos, etc.

El Tao de la Programación por Geoffrey James

LIBRO PRIMERO: EL VACÍO SILENCIOSO

Así habló el maestro programador: "Cuando hayas aprendido a extraer el código del error desde un trap frame, será la hora de marcharte"

1.1 Algo misterioso se forma, nace en el vacío silencioso. Esperando solo e inmóvil, al mismo tiempo detenido y en movimiento constante. Es la fuente de todos los programas. Yo no sé su nombre, así que lo llamaré el Tao de la Programación.

Si el Tao es grandioso, entonces el sistema operativo es grandioso. Si el sistema operativo es grandioso, entonces el compilador es grandioso. Si el compilador es grandioso, entonces la aplicación es grandiosa. El



usuario está complacido y hay armonía en el mundo.

El Tao de la Programación fluye lejos y regresa con el viento de la mañana.

1.2 El Tao engendró al lenguaje máquina. El lenguaje máquina dio vida al ensamblador. El ensamblador se la dio al compilador. Ahora hay diez mil lenguajes.

Cada lenguaje tiene su propósito, aunque sea humilde. Cada lenguaje expresa el Yin y el Yang del software. Cada lenguaje tiene su lugar dentro del Tao.

Pero no programes en COBOL si puedes evitarlo.

1.3 En el principio era el Tao. El Tao engendró el Espacio y Tiempo. Por tanto Espacio y Tiempo son el Yin y el Yang de la programación.

Los programadores que no comprenden el Tao siempre siempre se quedan sin tiempo y espacio para sus programas. Los programadores que comprenden el Tao siempre

tienen tiempo y espacio suficiente para lograr sus objetivos.

¿Cómo podría ser de otra manera?

1.4 Al programador sabio le hablan del Tao y lo sigue. Al programador medio le hablan del Tao y lo busca. El programador necio se ríe cuando le hablan del Tao.

Si no fuera por la risa, no existiría el Tao.

Los sonidos más altos son los más difíciles de oír.

Avanzar es un camino para la retirada.

El gran talento se muestra tarde en la vida.

Incluso un programa perfecto todavía tiene errores.

LIBRO SEGUNDO: LOS MAESTROS ANCIANOS
Así habló el maestro programador:

"Después de tres días sin programar, la vida pierde sentido"

2.1 Los programadores de la antigüedad eran misteriosos y profundos. No podemos comprender sus pensamientos, así que todo lo que hacemos es describir su apariencia.

El Tao de la programación

Consciente, cual zorro cruzando el agua. Alerta, como un general en el campo de batalla. Amable, como una anfitriona saludando a sus invitados. Simple, como bloques de madera sin tallar. Opaco, como negras piscinas en cuevas oscuras.

¿Quién puede contar los secretos de sus corazones y mentes?

La respuesta sólo existe en el Tao.

2.2 El Gran Maestro Turing una vez soñó que era una máquina. Cuando se despertó, exclamó:

"¡No sé si soy Turing soñando que soy una máquina, o una máquina soñando que soy Turing!"

2.3 Un programador de una gran compañía fue a una conferencia de software y luego regresó para informar a su jefe, diciendo: "¿Qué clase de programadores trabajan en otras empresas? Se comportan mal y no se preocupan por las apariencias. Su cabello era largo y despeinado y sus ropas arrugadas y viejas. Destrozaron nuestra hospitalidad e hicieron ruidos groseros durante mi presentación.

El director dijo: "Nunca debí haberte enviado a la conferencia. Esos programadores viven más allá del mundo físico. Consideran que la vida es absurda, una

coincidencia accidental. Ellos van y vienen sin conocer limitaciones. Sin cuidado, ellos viven sólo para sus programas. ¿Por qué deberían preocuparse por las convenciones sociales?

Ellos viven dentro del Tao".

2.4 El discípulo preguntó al Maestro:

"Este es un programador que nunca diseña, documenta o prueba sus programas. Sin embargo, todos los que lo conocen lo consideran uno de los mejores programadores del mundo. ¿Por qué es esto?"

El Maestro responde: "Ese programador ha alcanzado la maestría del Tao. Ha ido más allá de la necesidad de un diseño; no se enoja cuando el sistema se cae, pero acepta al universo sin preocupación. Ha ido más allá de la necesidad de documentación; no le importa si alguien más ve su código. Ha ido más allá de la necesidad de pruebas; cada uno de sus programas son perfectos en sí mismos, serenos y elegantes, su propósito es auto-evidente. Realmente, él ha penetrado en el misterio del Tao".

LIBRO TERCERO: DISEÑO
Así habló el maestro programador:

"Cuando el programa está siendo testeado, es demasiado



tarde para hacer hacer cambios de diseño"

3.1 Hubo una vez un hombre que fue a una feria de informática. Cada día, al entrar le decía al guarda de la puerta: "soy un gran ladrón reconocido por mis hazañas de robo. Estás prevenido de que esta feria no escapará sin ser saqueada".

Estas palabras incomodaron mucho al guardia, porque dentro había millones de dólares en equipamiento informático, así que observó cuidadosamente al hombre. Sin embargo, el hombre simplemente vagaba de stand en stand, murmurando para sí.

Cuando el hombre se iba, el guardia se lo llevó aparte y buscó entre sus ropas, pero nada fue encontrado.

Al siguiente día de la feria, el hombre regresó y regañó al guardia diciendo:

El Tao de la programación

"Ayer escapé con un gran botín, pero hoy será todavía mejor". Así que el guardia lo observó incluso más de cerca, pero sin resultados.

En el último día de la feria, el guardia no pudo resistir más su curiosidad. "Señor Ladrón", dijo, "estoy tan confuso que no puedo vivir en paz. Por favor ilumíneme. ¿Qué es lo que está robando?"

El hombre sonrió. "Estoy robando ideas", dijo.

3.2 Había una vez un maestro programador que escribía programas no estructurados. Un programador novicio, buscando imitarlo, también comenzó a escribir programas no estructurados. Cuando el novicio le pidió al maestro que evaluara su progreso, el maestro lo criticó por escribir programas no estructurados, diciendo:

"Lo que es apropiado para el maestro no es apropiado para los principiantes. Debes entender el Tao antes de trascender la estructura".

3.3 Hubo una vez un maestro programador en la corte del señor de Wu. El señor preguntó al programador: "¿qué es más fácil de diseñar, un paquete de contabilidad o un sistema operativo?"

"Un sistema operativo", respondió el programador.

El señor lanzó una exclamación de incredulidad. "Sin duda, un paquete de contabilidad es trivial al lado de la complejidad de un sistema operativo", dijo.

"No es así", dijo el programador, "cuando se diseña un paquete de contabilidad, el programador actúa como mediador entre personas con distintas ideas: cómo debe operar, cómo deben aparecer sus informes, y cómo se deben cumplir las leyes de impuestos". Por el contrario, un sistema operativo no está limitado por las apariencias externas. En el diseño de un sistema operativo, el programador busca la armonía más simple entre máquina e ideas. Esta es la razón por la que el sistema operativo es más fácil de diseñar".

El señor de Wu asintió y sonrió. "Eso está bien, pero, ¿qué es más fácil de depurar?"

El programador no respondió.

3.4 Un gerente fue al maestro programador y le mostró el documento de requisitos para una nueva aplicación. El gerente preguntó al maestro: "¿Cuánto tiempo se tarda en diseñar este sistema si le asigno cinco programadores?"

"Tomará un año", dijo el maestro rápidamente.

"¿Pero necesitamos este sistema inmediatamente o

incluso antes! ¿Cuánto se tarda si le asigno diez programadores?"

El maestro programador frunció el ceño. "En este caso, se tardará dos años".

"¿Y si le asigno cien programadores?"

El maestro programador se encogió de hombros. "Entonces el diseño no se completará jamás", dijo.



LIBRO CUARTO:
CODIFICACIÓN

Así habló el maestro programador:

"Un programa bien escrito es su propio cielo; un programa mal escrito, su propio infierno"

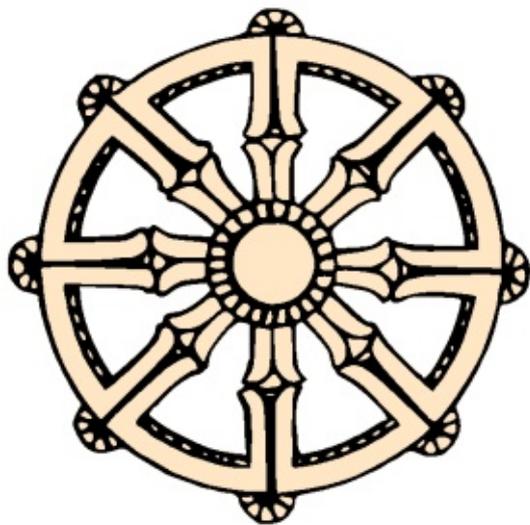
4.1 Un programa debe ser ligero y ágil, sus subrutinas conectadas como las perlas de un collar. El espíritu e intencionalidad del programa debe mantenerse en todo momento. No debe ser ni mucho ni poco, ni bucles innecesarios ni

El Tao de la programación

variables sin utilidad, ni ausencia de estructura ni rigidez excesiva.

Un programa debe seguir la "Ley de la menor sorpresa". ¿Qué es esta Ley? Simplemente que el programa debe responder siempre de la forma que menos sorprenda al usuario.

Un programa, no importa cuán complejo sea, debería actuar como una sola unidad. El programa debe ser dirigido por la lógica interna en lugar de por las apariencias externas.



Si el programa falla en estos requisitos, se llegará a un estado de desorden y confusión. La única manera de corregir esto es reescribiendo el programa.

4.2 Un discípulo preguntó al maestro: "Tengo un programa que a veces funciona y veces aborta. He seguido las reglas de la programación, y estoy totalmente desconcertado.

¿Cuál es la razón?".

El maestro respondió: "Estás confuso porque no entiendes el Tao. Sólo un necio espera un comportamiento racional de sus semejantes humanos. ¿Por qué ibas a esperarlo de una máquina que los humanos han construido? Los ordenadores simulan determinismo; sólo el Tao es perfecto.

Las reglas de la programación son transitorias; sólo el Tao es eterno. Por tanto, debes contemplar el Tao antes de ser iluminado."

"Pero, ¿cómo sabré que he sido iluminado?", preguntó el discípulo.

"Tu programa funcionará correctamente", respondió el maestro.

4.3 Un maestro estaba explicando la naturaleza del Tao a uno de sus discípulos. "El Tao está presente en todo el software, independientemente de su insignificancia", dijo el maestro.

"¿Está el Tao en una calculadora de bolsillo?", preguntó el novicio.

"Está", fue la respuesta.

"¿Está el Tao en un videojuego?", continuó el discípulo.

"Incluso en un videojuego",

dijo el maestro.

"¿Y está en el sistema operativo de un ordenador personal?"

El maestro tosió y cambió levemente de posición. "La lección ha acabado por hoy", dijo.

4.4 El programador del Príncipe Wang estaba codificando software. Sus dedos bailaban sobre el teclado. El programa compiló sin errores y se ejecutó cual ligera brisa.

"¡Excelente!", exclamó el Príncipe, "¡Tu técnica es infalible!".

"¿Técnica?", dijo el programador girándose desde su terminal, "¡Lo que yo sigo es el Tao más allá de toda técnica! Cuando empecé a programar, veía ante mí el problema completo como un todo.

Después de tres años ya no veía ese bloque: empecé a usar subrutinas. Pero ahora no veo nada. Mi ser existe en un vacío sin forma. Mis sentidos están ociosos. Mi espíritu, libre para trabajar sin un plan, sigue su propio instinto. En resumen, mi programa se escribe a sí mismo. Es cierto que a veces hay problemas complejos. Los veo acercarse, me detengo, observo en silencio. Entonces cambio una única línea de código y las dificultades

El Tao de la programación

se desvanecen como una voluta de humo. Compilo mi programa. Me quedo quieto y dejo que el gozo del trabajo llene mi ser. Cierro los ojos un momento, y entonces cierro mi sesión”.

El Príncipe Wang dijo “Ojalá todos mis programadores fueran tan sabios”.

LIBRO QUINTO:
MANTENIMIENTO
Así habló el maestro programador:

“Incluso un programa de tres líneas algún día tendrá que ser mantenido”

5.1 Una puerta bien usada no necesita aceite en sus bisagras.

Un río que fluye veloz no se estanca.

Ni el sonido ni los pensamientos pueden viajar a través del vacío.

El software se pudre si no se utiliza.

Son grandes misterios.

5.2 Un gerente preguntó a un programador que cuánto tiempo le llevaría terminar el programa en el que trabajaba. “Se acabará mañana”, respondió rápidamente el programador.

“Creo que no estás siendo realista”, dijo el gerente, “De verdad, ¿cuánto tiempo tardará?”.

El programador pensó un instante. “Tengo algunas

características que me gustaría añadirle. Me llevará al menos dos semanas”, dijo finalmente. “Incluso eso es demasiado esperar”, insistió el gerente,



“me basta si simplemente me avisas cuando el programa esté completo”.

El programador asintió.

Varios años más tarde, el gerente se retiró. De camino hacia su almuerzo de jubilación descubrió al programador dormido sobre su terminal. Había estado programando toda la noche.

5.3 Un programador novicio fue una vez asignado a la codificación de un sencillo paquete financiero.

El novicio trabajó furiosamente muchos días, pero cuando su maestro revisó su programa descubrió que

contenía un editor de pantallas, un conjunto de rutinas gráficas generales, y una interfaz de inteligencia artificial, pero ni la más mínima mención de nada financiero.

Cuando el maestro le preguntó acerca de ello, el novicio se indignó. “No seas tan impaciente”, dijo, “Incluiré los temas financieros al final”.

5.4 ¿Acaso un buen agricultor descuidaría un cultivo que ha plantado?

¿Acaso descuidaría un buen profesor incluso al estudiante más humilde?

¿Acaso un buen padre permitiría que uno de sus hijos murieran de hambre?

¿Acaso un buen programador rechazaría mantener su propio código?

LIBRO SEXTO: GESTIÓN
Así habló el maestro programador:

“Sean los programadores muchos y gestores pocos; todos ellos serán entonces productivos”

6.1 Cuando los gestores tienen reuniones infinitas, los programadores escriben juegos. Cuando los financieros hablan de beneficios cuatrimestrales, el presupuesto de desarrollo está a punto de ser recortado.

Cuando los científicos hablan de cielo azul, las nubes están a punto de aparecer.

El Tao de la programación

Ciertamente, esto no es el Tao de la Programación.

Cuando los gestores de comprometen, los juegos son ignorados. Cuando los financieros hacen planes a largo plazo, la armonía y el orden son restaurados. Cuando los científicos se centran en los problemas cercanos, los problemas estarán a punto de resolverse.

Ciertamente, esto es el Tao de la Programación.

6.2 ¿Por qué los programadores son improductivos?

Porque pierden su tiempo en reuniones.

¿Por qué los programadores son rebeldes?

Porque la gestión interfiere mucho.

¿Por qué los programadores reniegan unos de otros?

Porque están quemados.

Después de haber trabajado

para un mal gestor, ya no valoran sus empleos.

6.3 Un gerente estaba a punto de ser despedido, pero un programador que trabajaba para él inventó un nuevo programa que se hizo popular y se vendió bien. Como consecuencia, el gerente conservó su empleo.

El gerente intentó darle al programador una bonificación, pero éste se negó diciendo "yo escribí el programa porque pensé que era un concepto interesante, por lo que no espero ninguna recompensa".

Al oír esto, el gerente comentó: "Este programador, a pesar de su baja autoestima, entiende bien los deberes de un empleado. ¡Vamos a promocionarlo hacia la posición de consultor de gestión!".

Pero cuando se le dijo esto, el programador lo rechazó una vez más diciendo: "Vivo para la programación. Si fuera ascendido no haría más que desperdiciar el tiempo de todos. ¿Me puedo ir? Tengo un programa en el que trabajar".

6.4 Un gerente se dirigió a sus programadores: "En cuanto a sus horas de trabajo, van a tener que venir desde las nueve de la mañana hasta las cinco de la tarde". En ese momento todos se enfadaron y muchos de ellos renunciaron en el acto.

Así que el gerente dijo: "Bien, pues en ese caso podéis establecer vuestros propios horarios de trabajo, siempre que terminéis los proyectos a tiempo". Los programadores, ahora satisfechos, comenzaron a llegar a mediodía y trabajar hasta altas horas de la madrugada.

LIBRO SÉPTIMO:
CONOCIMIENTO
CORPORATIVO

Así habló el maestro programador:

"Puedes mostrar un programa a un ejecutivo de la empresa, pero no puedes hacerlo experto en informática"

7.1 Un discípulo preguntó al maestro: "En el Este hay una gran estructura con forma de árbol que los hombres llaman sede corporativa. Está excesivamente inflada con vicepresidentes y contables. Generan una gran cantidad de notas diciendo 've aquí' o 've allá' y nadie sabe lo que significa. Cada año se colocan nuevos nombres en las ramas, todo en vano. ¿Cómo puede existir una entidad tan innatural?".

El maestro respondió: "Percibes esta inmensa estructura y te perturba que no tenga un propósito racional. ¿No puedes encontrar entretenimiento con sus giros sin fin? ¿No disfrutas de la facilidad de programar sin problemas refugiado bajo sus ramas?"



El Tao de la programación



¿Por qué te molesta su inutilidad?”.

7.2 En el Este hay un tiburón que es mayor que todos los otros peces. Se transforma en ave cuyas alas son como nubes llenando el cielo. Cuando es ave, se mueve por toda la tierra y trae un mensaje desde la sede corporativa. Este mensaje cae entre los programadores como una gaviota dejando su huella en la playa. Entonces el pájaro remonta el vuelo y, con el cielo azul a sus espaldas, vuelve a casa.

El programador novicio mira sorprendido el ave porque no lo entiende. El programador intermedio teme la llegada del ave porque teme su mensaje. El maestro programador continúa trabajando en su terminal, no sabe que el ave ha llegado y se ha marchado.

7.3 El Mago de la Torre de Marfil llevó su último invento para que lo examinara el maestro programador. El Mago acarreó una gran caja negra a la oficina del maestro, mientras éste esperaba en

silencio.

“Esto es una estación de trabajo de propósito general integrada y distribuida”, comenzó el Mago, “diseñada ergonómicamente con un sistema operativo propietario, lenguajes de sexta generación y múltiples interfaces de usuario de tecnología punta. Construirlo costó a mis asistentes varios cientos de años/hombre. ¿No es sorprendente?”.

El maestro alzó sus cejas ligeramente. “Sin duda es increíble”, dijo.

“La sede corporativa ha ordenado”, continuó el Mago, “que todos usen esta estación de trabajo como plataforma para los nuevos programas. ¿Está de acuerdo con esto?”.

“Ciertamente”, respondió el maestro, “Lo transportaré al centro de datos inmediatamente”. Y el Mago retornó complacido a su torre.

Varios días después, un novicio vagaba por la oficina del maestro programador y le dijo: “No puedo encontrar el listado de mi nuevo programa. ¿Sabes dónde puede estar?”.

“Sí”, respondió el maestro, “los listados están apilados sobre la plataforma del centro de datos”.

7.4 El maestro programador se mueve de un programa a otro

sin miedo. Ningún cambio en los gestores puede dañarle. No será despedido, ni siquiera aunque el proyecto en el que trabaja sea cancelado. ¿Por qué es esto?

El Tao está en él.

LIBRO OCTAVO: HARDWARE Y SOFTWARE

Así habló el maestro programador:

“Sin el viento, el pasto no se mueve.

Sin software, el hardware es inútil”

8.1 Un discípulo preguntó al maestro: “Percibo que una compañía de ordenadores es mucho mayor que todas las demás. Se eleva por encima de su competencia como un gigante entre enanos. Cualquiera de sus divisiones podría abarcar todo el negocio. ¿Por qué es esto así?”.

El maestro respondió: “¿Por qué haces preguntas tan necias? Esa compañía es así de grande porque es grande. Si sólo fabricara hardware nadie lo compraría. Si sólo hiciera software, nadie lo usaría. Si sólo mantuviera sistemas, la gente los trataría como a sirvientes. Pero al combinar todas esas cosas, la gente piensa que son dioses. Al no buscar la confrontación conquista sin esfuerzo”.

8.2 Un maestro programador pasó un día junto a un novicio.

El Tao de la programación

El maestro notó la preocupación del novicio con un juego en un dispositivo portátil. "Disculpe", dijo, "¿me permite examinarlo?".

El novicio atendió y pasó el dispositivo al maestro. "Veo que el aparato afirma tener tres niveles de juego: fácil, intermedio y difícil", dijo el maestro. "Pero aún cada dispositivo tiene otro nivel de juego, donde el apartado no busca conquistar al humano, ni ser conquistado por el humano".

"Ruego, gran maestro", imploró el novicio, "¿cómo hace uno para encontrar esa misteriosa configuración?".

El maestro arrojó el dispositivo al suelo y lo aplastó bajo su pie. Y de pronto, el novicio fue iluminado.

8.3 Había una vez un programador que trabajaba con microordenadores. "Mira lo bien que estoy aquí", dijo a un programador de mainframes que lo fue a visitar. "Tengo mi propio sistema operativo y dispositivo de almacenamiento de archivos. No tengo que compartir mis recursos con nadie. El software es consistente y fácil de usar. ¿Por qué no dejas tu trabajo

actual y te vienes conmigo?"

Entonces, el programador de mainframes comenzó a describir su sistema a su amigo, diciendo: "El mainframe está sentado como un antiguo sabio meditando en el centro de datos.

Sus discos se encuentran de extremo a extremo como un gran océano de maquinaria. El software es tan polifacético como un diamante, y enrevesado como una selva virgen. Los programas, cada uno único, se mueven a través del sistema como un río de corriente rápida. Por eso estoy feliz donde estoy".

El programador de microordenadores, al oír esto, se quedó en silencio. Pero los dos programadores siguieron siendo amigos hasta el final de sus días.

8.4 Hardware y Software se encontraron en el camino hacia Changtse. Software dijo: "Tú eres Yin y yo soy Yang. Si viajamos juntos nos haremos famosos y ganaremos vastas sumas de dinero". Y así, el equipo se unió, pensando que conquistarían el mundo.

Actualmente se encontraron con Firmware, que estaba vestido con harapos y cojeaba apoyado en un palo espinoso. Firmware les dijo: "El Tao está más allá del Yin y Yang. Es silencioso y quieto como un estanque de agua. No busca la fama, por tanto, nadie sabe de su presencia. No busca fortuna, porque es completo en sí mismo. Existe más allá del espacio y del tiempo".

Software y hardware, avergonzados, regresaron a sus hogares.

LIBRO NOVENO: EPÍLOGO
Así habló el maestro programador:

"Es hora de que partas"

safe creative 



VeneHosting.com

Tu Hosting en Venezuela