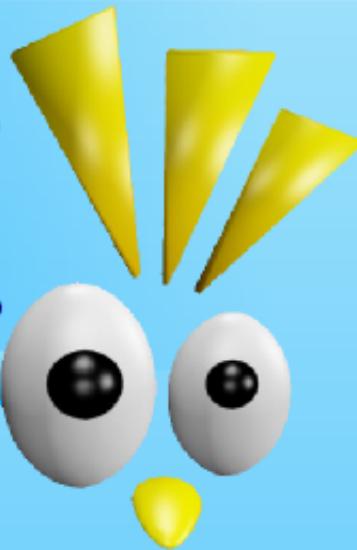


Linvix



Compilando nuevo kernel en Ubuntu.

Zentyal. El Servidor de las PYME.

Instalación Servidor OpenVPN y cliente Windows.

Introducción a Perl.

CUPS. Instalar una impresora Epson LX 300+ en Ubuntu.

y más...



Editorial

Staff y Coordinación

Roylán Suárez Reyes
Javier Suárez Rodríguez
Osniel Cabeza García

Diseño

Javier Suárez Rodríguez

Revisión y Corrección

Milaydis Rodríguez Martín

Contacto

linvix@gmail.com

Renuncia de Responsabilidad

Todos los artículos, noticias y comentarios publicados en Linvix son propios de los autores. Los contenidos que se ofrecen han sido probados por el autor, por lo que Linvix no se responsabiliza con los daños o pérdida de información que lleven la realización de alguno de los artículos publicados.



Una vez más y casi cumpliendo el segundo aniversario llega Linvix 11, una revista de software libre y código abierto que se realiza gracias al esfuerzo de la comunidad alrededor de todo el mundo y que hace que no solo este proyecto, sino muchos otros existan hoy.

No solo estamos contentos porque estemos arribando el segundo aniversario, sino porque además estrenamos una nueva casa: linvix.espaciolinux.com, todo el agradecimiento para Paco Revilla, Director General de Espacio Linux, quien gentilmente donó el hosting para hacer posible que este proyecto cumpla con los objetivos trazados y continúe brindando un servicio que satisfaga a nuestros lectores.

Esta nueva edición viene cargada de artículos interesantes, dos artículos sobre el Kernel de Linux que te enseñarán a compilar el kernel en Ubuntu, iniciamos dos nuevas series de artículos, una de programación en Perl y otra sobre Zentyal, a las cuales le daremos continuidad en próximos números de Linvix, además de las acostumbradas secciones de la revista.

También queremos invitarlos a todos a participar en la Encuesta **Linvix 2011**, que está disponible en nuestro sitio web para conocer mejor a nuestros lectores y así enfocarnos en un mejor desarrollo de la revista. La encuesta estará disponible hasta el 30 de marzo y los resultados serán publicados en la edición del segundo aniversario de la revista.

Si eres uno de nuestros asiduos lectores seguramente te habrás dado cuenta que hemos demorado en publicar la edición número 11. Este número se ha realizado a puro pulmón, en primer lugar Linvix ha sufrido cambios en el equipo de trabajo y en el modo de realización, pues dos miembros -Danelia y Fernando- se nos han separado por un tiempo y en segundo lugar hemos tenido que hacer todo el trabajo de edición y maquetación en la casa de nuestro amigo Killer así que por lo visto hemos realizado un esfuerzo enorme para cumplir con el compromiso de hacer una entrega cada dos meses.

Como siempre queremos agradecer a toda la comunidad que hace posible la existencia de este proyecto, pero en especial a Paco por habernos brindado un espacio casi desde los mismos inicios de Linvix, a Killer por habernos permitido invadir su casa durante el fin de semana, y a xdad por permitirnos realizar la publicación de Linvix 11.

El equipo de Linvix

Contenido

Pág 4

Comunidad

Pág 6

Laboratorio

Parche, Compilar e Instalar un Módulo slamr del Módem PCI TP-LINK Motorola SM56 Data Fax para Kernel Linux 2.6.36.

Compilando nuevo kernel en Ubuntu.

Pág 14

La Distro

Zentyal. El Servidor de las PYMEs.

Pág 21

Servidores

Instalación Servidor OpenVPN y cliente Windows.

Pág 30

Programación

Introducción a Perl.

Pág 33

How-To

CUPS. Instalar una impresora Epson LX 300+ en Ubuntu.

Cómo regular el envío y recepción de correos electrónicos a través de conexiones PPP.

Cómo cifrar y utilizar el directorio Private en Ubuntu.

Pág 39

Entrevista

Entrevista a Darien Alonso Camacho. Proyecto Cazabe.

Pág 42

Noticias

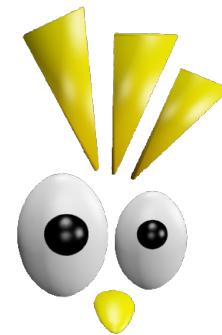
Pág 45

El Personaje

Richard Matthew Stallman

Comunidad

FAQ Linvix



1 - ¿Qué es Linvix?

- Linvix es una Revista de Software Libre y Código Abierto.

2 - Objetivos de la Revista

- Tiene como objetivo principal promover y divulgar el uso del Software Libre en nuestro municipio.
- Transmitir nuestro conocimiento.
- Dar soporte y ayudar a todos aquellas personas y/o instituciones interesadas en el uso del Software Libre.

3 - ¿Cómo llega Linvix a sus lectores?

- En formato digital a través de nuestro sitio web.

4 - ¿Cómo obtener número anteriores?

- Todos los números están disponibles para su descarga desde:

linvix.espaciolinux.com

5 - ¿Con qué frecuencia se publica Linvix?

- Cada dos meses

6 - ¿Quiénes realizan Linvix?

- Es realizada por un pequeño grupo de usuarios apasionados al Software Libre y el Código Abierto del Municipio de Viñales, provincia Pinar del Río, Cuba, con la colaboración desinteresada de muchas personas del resto del país y el mundo.

7 - ¿Puedo colaborar con la revista?

- Estamos abiertos a todas aquellas personas que deseen colaborar con la revista. Nos pueden enviar sus artículos, opiniones, sugerencias, etc.... a nuestra dirección de correo electrónico linvix@gmail.com

NOTA: Los artículos no deben sobrepasar de 1 MB.

Comunidad

¿Ya eres



fan de Linvix?

Estamos en

<http://twitter.com/linvix>



Subscríbete a nuestra lista de discusión Linvix

<http://groups.google.com.cu/group/linvix>

Para Subscribirse envíe un correo a:
linvix-subscribe@googlegroups.com



Laboratorio

Parche, Compilar e Instalar Módulo slamr del Módem PCI TP-LINK Motorola SM56 Data Fax para Kernel Linux 2.6.36

Oleksis Fraga Menéndez

Universidad de Matanzas "Camilo Cienfuegos"

Facultad de Ingeniería Informática

ofraga@ipimtzag.rimed.cu

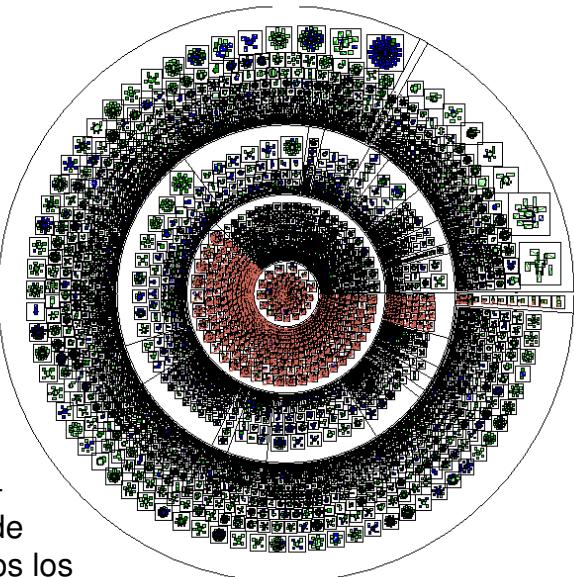
En el artículo Configurar, Compilar e Instalar Kernel de Linux y drivers para Modems PCI en Debian GNU/Linux 5.0 "Lenny" del número anterior de Linvix, vimos como se realizaba este proceso para los kernel de Linux anteriores al 2.6.36.

Al realizar la actualización del kernel a la versión 2.6.36 y compilar el módulo slamr (slmodem-2.9.11-20100718) recibimos un error de compilación. Para solucionar el error la comunidad de linmodems.org dispone de un parche para este driver. Para su instalación realizamos los siguientes pasos:

El slmodem-2.6.36.patch tiene el siguiente contenido:

```
--- slmodem-2.9.11-20100718/drivers/amrmo_init.c.orig      2010-10-10 00:33:03.000000000 -0400
+++ slmodem-2.9.11-20100718/drivers/amrmo_init.c      2010-10-10 00:37:36.000000000 -0400
@@ -436,8 +436,11 @@
}

-
+[#if LINUX_VERSION_CODE >= KERNEL_VERSION(2,6,36)
+static long amrmo_ioctl(struct file *file, unsigned int cmd, unsigned long parg)
+#else
static int amrmo_ioctl(struct inode *inode, struct file *file, unsigned int cmd, unsigned long parg)
+#endif
{
    struct amrmo_struct *amrmo = (struct amrmo_struct *)file->private_data;
    unsigned long flags;
@@ -514,7 +517,11 @@
```



Laboratorio

```
.read = amrmo_read,  
.write = amrmo_write,  
.poll = amrmo_poll,  
.ioctl = amrmo_ioctl,  
+ #if LINUX_VERSION_CODE >= KERNEL_VERSION(2,6,36)  
+ .unlocked_ioctl = amrmo_ioctl,  
+ #else  
+ .ioctl = amrmo_ioctl,  
+ #endif  
.open = amrmo_open,  
.release = amrmo_release,  
};
```

Para aplicar el parche, ejecutamos lo siguiente:

```
# cp slmodem-2.6.36.patch slmodem-2.9.11-20100718  
# cd slmodem-2.9.11-20100718  
# patch -p1 < slmodem-2.6.36.patch
```

y nos debe mostrar la salida:
patching file drivers/amrmo_init.c

Compilamos slamr para kernel 2.6.36

```
# make  
# make install
```

Con esto configuramos, compilamos e instalamos el módulo para el kernel linux 2.6.36. Gracias a todos los usuarios de la comunidad y puedes encontrar otros recursos en <http://linmodems.technion.ac.il>

Participa en la Encuesta Linvix 2011 y ayudanos a conocer mejor a nuestros lectores.



Laboratorio

Compilando nuevo kernel en Ubuntu.

Lic. Ixen Rodríguez Pérez
GNU/Linux Proud User #313158
tuxisla@ij.cc.cu

Antes de adentrarnos en los aspectos técnicos de la compilación del kernel Linux, conocer un poco de historia es necesario.

Debemos comenzar por respondernos la pregunta ¿qué es el kernel? El kernel en sí es el corazón o núcleo del Sistema Operativo. En nuestro caso, el kernel se llama Linux, y el SO lo componen el kernel junto con una serie de programas y aplicaciones del sistema GNU, es por ello que debemos llamar GNU/Linux al SO presente en las distribuciones que tan cotidianamente utilizamos.

Una pregunta que a veces los usuarios principiantes se hacen, es justamente esta: ¿para qué compilar el kernel?, si ya tienen su máquina funcionando. La respuesta es simple, y es que entre una versión y otra de los distintos kernel se agregan características nuevas al mismo, además de una serie de mejoras. Por otro lado, el kernel que acompaña a las distribuciones es un kernel genérico, esto quiere decir que no tiene ningún tipo de optimización para nuestro hardware específico, viene por defecto para procesadores i386, y lo más probable es que contenga soporte innecesario para una gran cantidad de dispositivos, que no poseemos o no utilizamos.

Por último, compilar el kernel es un paso altamente educativo y didáctico para comprender en mayor profundidad el funcionamiento del SO GNU/Linux, y siempre es bueno saberlo por si alguien pregunta.

Luego, ¿qué necesito para poder compilar el kernel? En primer lugar, un compilador gcc. Necesitamos además las librerías de desarrollo



correspondientes al sistema glibc, y en algunos casos requeriremos también del ensamblador que se distribuye en el paquete binutils. Otros ensambladores disponibles para GNU/Linux son gasm o nasm. Una vez que confirmemos la existencia de todas estas herramientas en nuestro sistema, es necesario contar, obviamente, con el código fuente del kernel.

En la mayoría de las distribuciones se incluye el código fuente en los discos de instalación, pero no se instalan por defecto. De todas maneras es recomendable bajar siempre la última versión puesto que la mayoría de los fabricantes de distribuciones incluyen código adicional en su kernel, que no forma parte del original.

Si tenemos la seguridad que contamos con los fuentes 100% originales sin modificaciones, es posible actualizar a la última versión a través de parches. Los parches son archivos que contienen las diferencias entre un árbol de fuentes y otro. Por ende, es mucho más conveniente actualizar nuestro kernel a través de este método sobre todo si contamos con

Laboratorio

una conexión a Internet relativamente lenta, debido a que los parches son de muy pequeño tamaño.

Ahora que ya conocemos el por qué debemos compilar nuestro sistema con la última versión posible del código fuente del kernel Linux, estamos en condiciones de, con mucha paciencia, poner manos a la obra.

Este documento se basa en la traducción del artículo Kernel Howto, publicado en inglés en la Wiki de Ubuntu; resumiendo algunos puntos, enriqueciendo y aclarando otros. Y por supuesto, comprobando que funciona. La compilación del kernel se hace a la manera Debian, es decir, con herramientas propias de esta distribución. Vale destacar que los pasos descritos en los puntos 4.1. y 5.1., son aplicables a otras distribuciones como CentOS y RHEL.

Así que allá vamos, paso a paso:

1. Instalar paquetes necesarios.

Para compilar un kernel a medida vamos a necesitar los paquetes build-essential, kernel-package y otros, por lo tanto:

```
tuxisla@legolas:$ sudo aptitude install build-essential kernel-package libncurses5 libc6-dev dialog module-assistant
```

2. Obtener código fuente del kernel.

El código fuente del kernel lo podemos obtener de muchos sitios. El oficial, es <http://www.kernel.org/>. Pero también está en los repositorios de Ubuntu y a los efectos de este manual, se ha utilizado el que aparece en el repositorio porque tiene algunas obvias ventajas, como por ejemplo el comando make-kpkg, que humaniza la construcción de paquetes .deb para núcleos a partir de los fuentes. Sin embargo, si desea compilar el kernel utilizando los fuentes del sitio oficial, también aquí se detallan los pasos a seguir.

En los repositorios de Ubuntu encontraremos paquetes llamados kernel-source y linux-source. Ambos son el código fuente del kernel. ¿Cuál es la diferencia?, pues que los kernel-source vienen de Debian, y los linux-source vienen de Ubuntu. Por lo tanto para obtener el código fuente del kernel que tenemos instalado y corriendo en nuestro sistema, sólo ejecutamos:

```
tuxisla@legolas:$ sudo aptitude install linux-source
```

Con esto se descargará el archivo linux-source-2.6.version.tar.bz2 en el directorio /usr/src. A partir de ahora asumiremos que hemos descargado el linux-source-2.6.32-25, que es el último disponible para Ubuntu Lucid Lynx. Nos movemos a ese directorio:

```
tuxisla@legolas:$ cd /usr/src
```

Si existe un enlace llamado linux a un kernel antiguo, lo borramos:

```
tuxisla@legolas:/usr/src$ sudo rm -R linux
```

Laboratorio

Ahora vamos a descomprimir el archivo del código fuente:

```
tuxisla@legolas:/usr/src$ sudo tar jxvf linux-source-2.6.32-25.tar.bz2
```

Esto creará un directorio llamado linux-source-2.6.32-25. Creamos un enlace simbólico llamado linux a ese directorio:

```
tuxisla@legolas:/usr/src$ sudo ln -s linux-source-2.6.32-25 linux
```

3. Configurar el kernel.

Hay varias formas de configurar el kernel. Probablemente la mayoría de la gente utiliza una de estas dos: make menuconfig o make xconfig. La diferencia es que con make menuconfig no hace falta tener el entorno gráfico corriendo, y con make xconfig sí. Para utilizar make menuconfig necesitaremos el paquete libncurses5-dev (con todas sus dependencias) y para utilizar make xconfig necesitaremos el paquete libqt3-mt-dev y todas sus dependencias. Así que, según lo que queramos utilizar, tendremos que hacer:

```
tuxisla@legolas:/usr/src$ sudo aptitude install libncurses5-dev
```

o bien:

```
tuxisla@legolas:/usr/src$ sudo aptitude install libqt3-mt-dev
```

Una duda que siempre surge para configurar el kernel, es que si se puede partir de una configuración conocida. La respuesta es sí. Si tenemos, por ejemplo, instalado el kernel que trae Ubuntu por defecto (2.6.32-25-generic en Lucid), en el directorio /boot nos encontraremos con un archivo llamado config-2.6.32-25-generic, y ¿qué es ese archivo?, pues no es más que un archivo de texto con la configuración actual del kernel, por lo tanto, para partir de esa base y comenzar la configuración del nuevo kernel desde la actual, lo que tenemos que hacer es copiar dicho archivo al directorio /usr/src/linux-source-2.6.32-25, pero llamándolo .config; lógicamente tendremos que cambiar los números de versión del kernel por los que se ajusten a cada caso particular, para ello:

```
tuxisla@legolas:/usr/src$ cd linux/
```

```
tuxisla@legolas:/usr/src/linux$ sudo cp /boot/config-2.6.32-25-generic .config
```

Pues ahora ya podemos hacer, situados en el directorio /usr/src/linux, un sudo make oldconfig menuconfig o un sudo make oldconfig xconfig y modificar las opciones que queramos del kernel, es decir, configurarlo a nuestro gusto: tipo de CPU, soporte de memoria, controladores Wi-Fi; conociendo el hardware existente en nuestro ordenador, pero con mucho cuidado, eso sí, porque este es el paso crítico. Aquí es donde lo podemos arruinar todo si, por ejemplo, no incluimos algún módulo necesario para nuestro sistema. Para conocer sobre nuestro hardware podemos leer el manual del motherboard, o bien utilizar los comandos hardinfo, sysinfo, lspci o lshw, este último nos brinda la posibilidad de obtener un archivo html con las especificaciones técnicas de nuestro hardware, ejemplo: sudo lshw -html > Escritorio/my_hardware.html

Laboratorio

4. Construir el paquete del kernel.

Para construir el paquete con el que instalaremos el nuevo kernel vamos a usar el comando make-kpkg. Mediante el cual básicamente sustituimos los clásicos make dep, make clean, make bzImage y make modules (utilizados cuando compilábamos para Red Hat y sus derivados). Admite numerosas opciones y modificadores; ya saben: man make-kpkg, pero para lo que nos ocupa ahora, sólo necesitaremos hacer lo siguiente: make-kpkg clean y make-kpkg --append-to-version=.#### --initrd kernel_image kernel_headers.

Como se puede apreciar, tenemos que modificar ".####", que representa una secuencia alfanumérica a decisión propia, y que se añadirá al número de versión del kernel para diferenciar el kernel que estamos compilando de cualquier otro que ya esté instalado en el sistema. Por lo tanto .#### lo deberemos sustituir por lo que queramos, intentando no utilizar el guión de subrayado bajo (_) ni expresiones como "-386", "-686", "-K7", "-sparc", etc., porque expresiones similares son las usadas por los kernel precompilados de Debian y podrían inducir a error. Personalmente yo uso la fecha en la que compilo el kernel, de forma que si lo compilase hoy, 15 de diciembre de 2010, quedaría así: version=.151210.

Ahora, hay que hablar un poco sobre la opción --initrd. Esta opción lo que hace es crear una imagen initrd en el directorio /boot. Realmente no es imprescindible utilizarla siempre que a la hora de configurar el kernel hayamos incluido de forma estática y no como módulos, los controladores para al menos nuestro bus, controladores de discos y sistema de archivos de nuestro directorio raíz. Pero, y esto es sólo una experiencia personal, siempre que he intentado compilar un kernel sin esa opción, me he encontrado con un sistema que no arrancaba, así que yo la uso siempre. ¿Qué problema tiene?, pues que para que funcione correctamente la opción --initrd, hay que aplicar al código fuente del kernel el parche cramfs. Tranquilos, porque todos los kernel de Debian y por supuesto también los de Ubuntu, tienen ese parche ya aplicado. Pero es algo a tener en cuenta si decidieron obtener el código fuente del kernel del sitio oficial.

Bien, como he dicho antes, los comandos que tenemos que ejecutar son:

```
tuxisla@legolas:/usr/src/linux$ sudo make-kpkg clean
```

```
tuxisla@legolas:/usr/src/linux$ sudo make-kpkg --initrd --append-to-version=.151210 --revision=$(version) kernel_image kernel_headers
```

Tras ejecutar el segundo, nos saldrá un mensaje de aviso diciéndonos que la opción --initrd puede no funcionar como nosotros esperamos, y preguntándonos si queremos abortar el proceso. Como ya sabemos que a nosotros nos va a funcionar correctamente, contestamos que no.

4. 4.1. Recomilar desde código fuente del sitio oficial (**linux-2.6.36.2.tar.bz2**).

Con las opciones necesarias para nuestro sistema ya guardadas en el fichero .config y de regreso en la consola, hacemos una limpieza de la configuración actual usando sudo make mrproper, y posteriormente ejecutamos:

```
tuxisla@legolas:/usr/src/linux$ sudo make clean bzImage modules modules_install
```

Laboratorio

Para implementar opciones de seguridad adicionales al kernel, debemos descargar el parche correspondiente del sitio Web del proyecto grsecurity. Este proyecto, provee enfoque innovador para la seguridad global del sistema mediante la utilización de modelos de detección, preventión y contención multicapas. En nuestro caso utilizaremos el parche grsecurity-2.2.1-2.6.36.2-201012121726.patch, correspondiente a la gama estable del kernel linux-2.6.36.2. Una vez descargado, parcheamos el código fuente en el directorio /usr/src, para ello:

```
tuxisla@legolas:/usr/src$ sudo patch -p0 < ./grsecurity-2.2.1-2.6.36.2-201012121726.patch
```

Volvemos a editar la configuración del kernel ejecutando sudo make mrproper xconfig o sudo make mrproper menuconfig; salvamos los cambios en el fichero .config y estaríamos en condiciones de ejecutar nuevamente:

```
tuxisla@legolas:/usr/src/linux$ sudo make clean bzImage modules modules_install
```

Y ahora a esperar, porque el proceso de compilación puede durar mucho, dependiendo del CPU y memoria RAM disponibles.

5. Instalar el nuevo kernel.

Si todo lo anterior ha ido bien, en el directorio /usr/src se deben haber creado los archivos kernel-image-2.6.32-25.151210_2.6.32-25.151210_i386.deb y kernel-headers-2.6.32-25.151210_2.6.32-25.151210_i386.deb. Los cuales contienen los módulos, las cabeceras y el kernel compilado listo para instalar, lo que faremos con:

```
tuxisla@legolas:/usr/src$ sudo dpkg -i kernel-*.151210_i386.deb
```

Estos comandos son equivalentes al clásico make install, que a su vez coloca en su sitio la imagen initrd generada y actualiza el gestor de arranque grub. Luego, y por seguridad, borramos el enlace que habíamos creado:

```
tuxisla@legolas:/usr/src$ sudo rm -R linux
```

5. 5.1. Instalar desde el código fuente (linux-2.6.36.2.tar.bz2).

Para instalar el kernel basta con ejecutar sudo make install, pero si prefieren hacer todo manualmente y entender un poco mejor cómo funciona el proceso, entonces sigan las instrucciones que se dan a continuación; en caso contrario y asumiendo que el comando anterior funcionó, pueden ir directamente al final del documento.

Primero debemos copiar la imagen generada y los archivos .config y System.map al directorio /boot del sistema. La ubicación original de la imagen depende de la arquitectura. En nuestro caso hemos utilizado x86:

```
tuxisla@legolas:/usr/src/linux$ sudo cp arch/x86/boot/bzImage /boot/vmlinuz-2.6.36.2-grsec && sudo cp System.map /boot/System.map-2.6.36.2-grsec && sudo cp .config /boot/config-2.6.36.2-grsec
```

Laboratorio

Creamos la imagen de inicio del kernel:

```
tuxisla@legolas:/usr/src/linux$ sudo mkintrramfs -o /boot/initrd.img-2.6.36.2-grsec /lib/modules/2.6.36.2-grsec/
```

Actualizamos los enlaces simbólicos de la versión del kernel:

```
tuxisla@legolas:/boot$ ln -sf /boot/vmlinuz-2.6.36.2-grsec /vmlinuz && ln -sf /boot/initrd.img-2.6.36.2-grsec /initrd.img
```

Solo falta editar el fichero /boot/grub/grub.cfg (/boot/grub/menu.lst, en versiones anteriores a Lucid), para que nuestro nuevo kernel esté disponible en la lista de inicialización del grub:

```
tuxisla@legolas:/boot$ sudo vi grub/grub.cfg
```

```
title Ubuntu 10.04, kernel 2.6.36.2-grsec
uuid bb504ec2-2672-494f-b21b-d3cb4710737e
kernel /vmlinuz-2.6.36.2-grsec root=UUID=a8931008-33c6-4310-a452-45f4f1cda209 ro splash
initrd /initrd.img-2.6.36.2-grsec
quiet
```

O simplemente ejecutamos los comandos sudo update-initramfs y sudo update-grub que permiten al grub realizar automáticamente los cambios pertinentes en el menú de arranque del sistema:

```
tuxisla@legolas:/usr/src/linux$ sudo update-initramfs -c -k 2.6.36.2-grsec && sudo update-grub
```

Y ¡voilà! Finalmente reiniciamos el ordenador con el nuevo kernel y verificamos la versión en ejecución.

```
tuxisla@legolas:~$ uname -a
```

```
Linux legolas.linuxij.cu 2.6.32-25.151210 #1 SMP Wed Dec 15 10:29:45 UTC 2010 i686 GNU/Linux
```

o (en el caso del kernel del sitio oficial)

```
Linux legolas.linuxij.cu 2.6.36.2-grsec #1 SMP Wed Dec 15 10:29:45 UTC 2010 i686 GNU/Linux
```

Referencias

- <http://grsecurity.net/>
- <http://www.kernel.org/>
- [http://es.wikipedia.org/wiki/Núcleo_\(informática\)](http://es.wikipedia.org/wiki/Núcleo_(informática))
- <http://es.wikipedia.org/wiki/GNU/Linux>
- <http://www.ubuntulinux.org/wiki/KernelHowto>

Zentyal. El Servidor de las PYME.

En el número anterior de Linvix, en la sección Testimonios, tuve la oportunidad de compartir con ustedes la experiencia que vivimos durante la instalación del servidor de nuestro centro de trabajo, y las razones por las cuales tomamos la decisión de migrar de Debian Lenny a Zentyal.

En esta ocasión ampliaremos un poco más el tema para aquellos que se quedaron con deseos de conocer en profundidad esta magnífica distribución, haremos un recorrido desde la instalación hasta la puesta en marcha de las interfaces de red.

Zentyal (anteriormente conocido como eBox Platform) es un servidor de red unificada de código abierto para las PYMEs. Zentyal puede actuar gestionando la infraestructura de red, como puerta de enlace a Internet (Gateway), gestionando las amenazas de seguridad (UTM), como servidor de oficina, como servidor de comunicaciones unificadas o una combinación de éstas. Además, Zentyal incluye un marco de desarrollo (un framework) para facilitar el desarrollo de nuevos servicios basados en Unix.

Zentyal 2.0 fue publicado en septiembre 2010 y dispone de las siguientes características:

- * Gestión de redes
 - o Cortafuegos y encaminamiento
 - o Infraestructura de red
 - o Soporte de redes privadas virtuales
 - o Proxy HTTP
 - o Sistema de detección de intrusos
 - o Servidor de correo
- * Webmail
- * Servidor web
- * Autoridad de Certificación
- * Trabajo en grupo



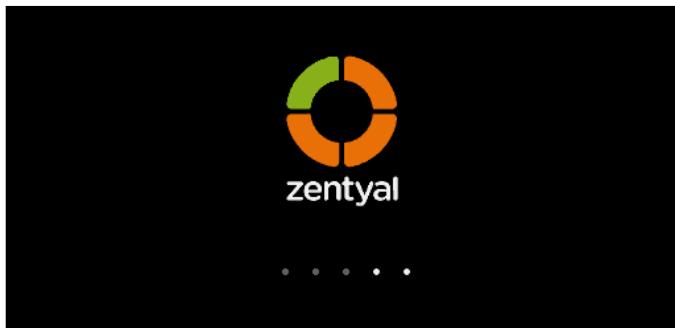
- o Gestión centralizada de usuarios y grupos
- o Controlador Primario de Dominio (PDC) de Windows
- o Compartición de recursos
- o Groupware: Compartición de calendarios, agendas, webmail, wiki, etc.
- o Servidor VozIP
- * Servidor Jabber/XMPP
- o Salas de conferencias
- * Rincón del Usuario de Zentyal
- * Informes y monitorización
 - o Dashboard para centralizar la información de los servicios
 - o Monitorización del CPU, carga, espacio del disco, temperatura, memoria
 - o Estado del RAID por software e información del uso de disco duro
 - o Informes completos y resumidos de sistemas
 - o Notificación de eventos vía correo, suscripción de noticias (RSS) o Jabber/XMPP
- * Actualizaciones de software
- * Copias de seguridad (backup de configuración y remoto de datos)

Zentyal 2.0 tiene como base la versión 10.04 de Ubuntu ya que esta es LTS y posee soporte por 3 años, así que el proceso de instalación es tan sencillo como instalar Ubuntu o Debian...

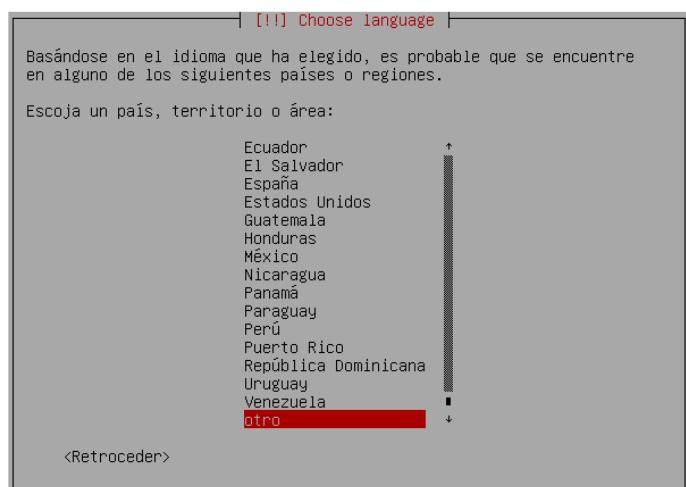
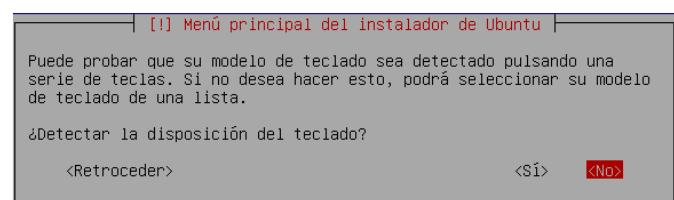
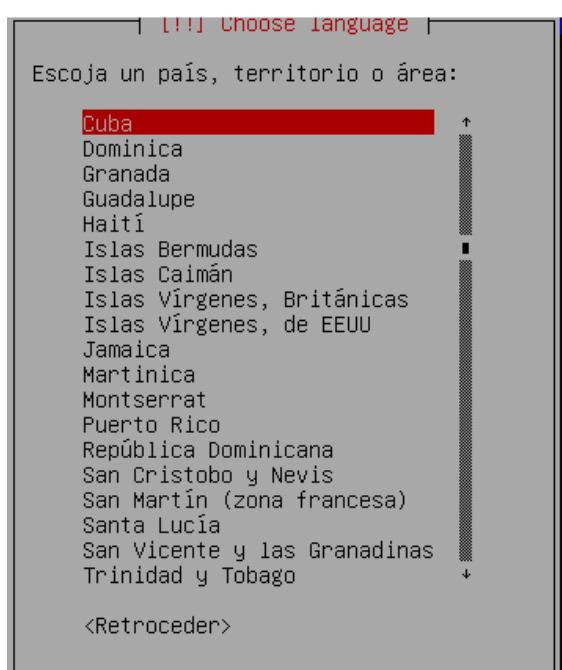
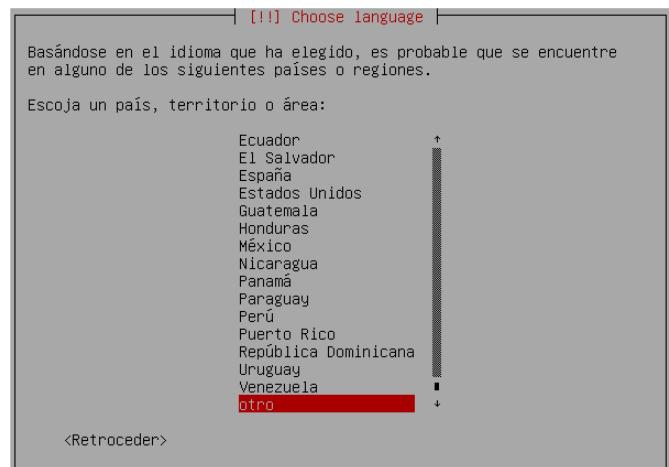
Introducimos el CD de instalación en el lector de CD y reiniciamos la PC, nos aseguramos que el SETUP esté configurado correctamente para que el booteo se realice por el lector de

La Distro

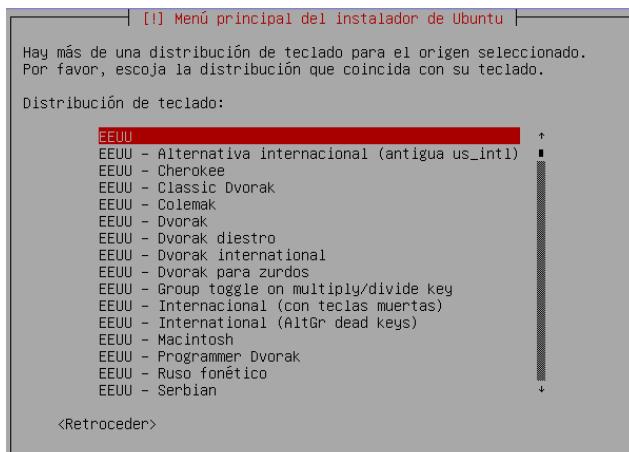
cd y veremos la primera imagen del proceso de instalación



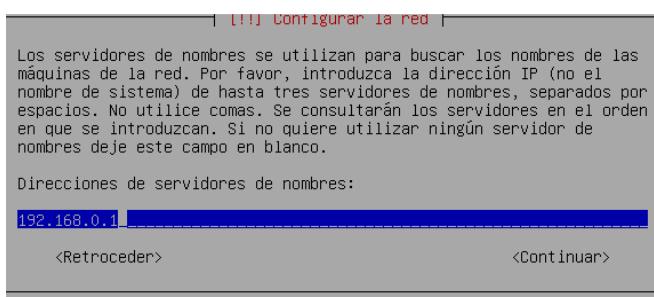
Posteriormente elegimos si queremos detectar la disposición del teclado, si conocemos bien cual es nuestro tipo elegimos NO y seleccionamos en la lista nuestro modelo.



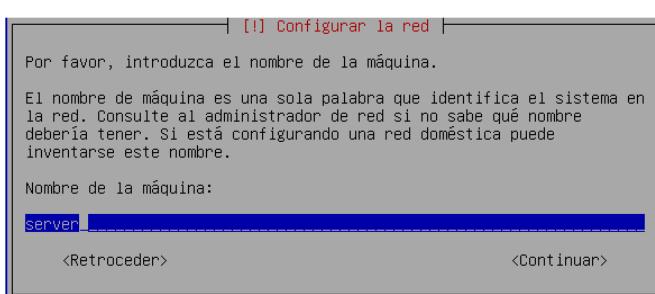
La Distro



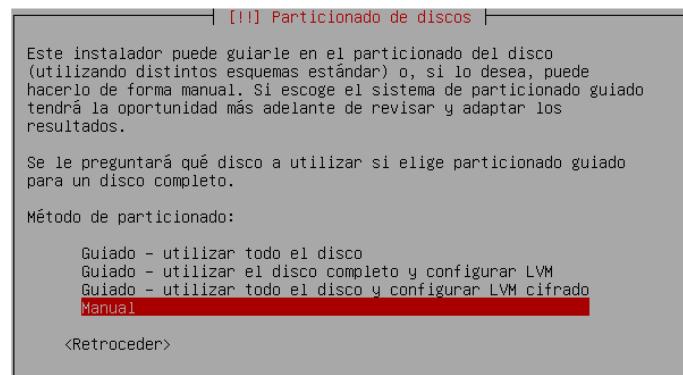
Ahora pasamos a configurar la red, lo primero es establecer los servidores de nombres, podemos introducir hasta tres direcciones ip separadas por espacio.



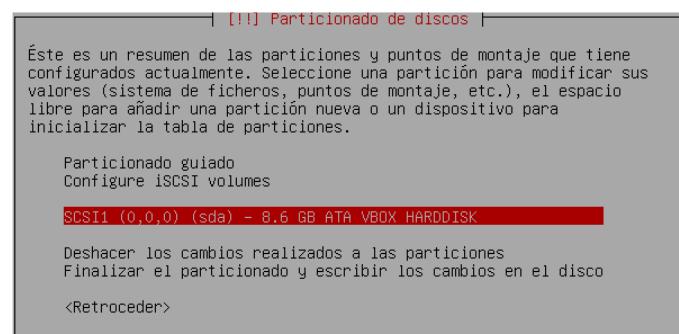
En el próximo paso elegimos el nombre de la máquina, es este caso Server pero puede ser cualquier otro nombre que no contenga espacios ni caracteres no alfanuméricos.



Configurada la red procedemos a realizar el particionado, uno de los pasos más difíciles sobre todo para los usuarios noveles, no porque realmente sea difícil, sino porque desconocen el funcionamiento del mismo. Seleccionamos el método manual.



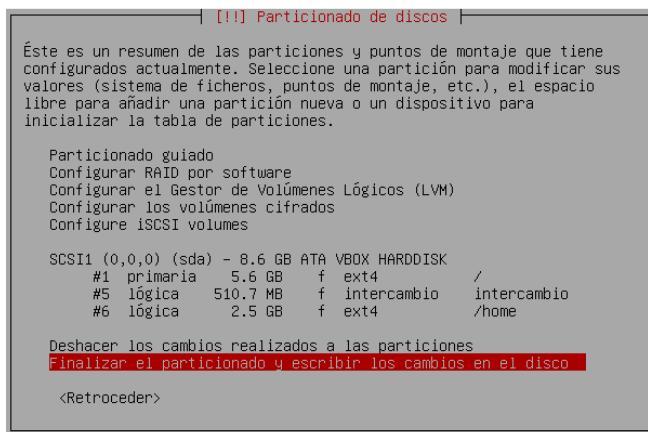
En próximo paso observamos un resumen de las particiones y puntos de montaje que tenemos configurados actualmente, nos ubicamos encima de la partición o disco sobre la cual deseamos realizar la instalación y presionamos Enter



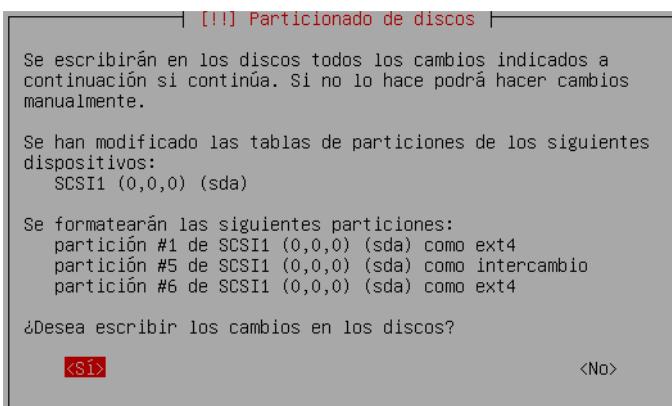
Realizamos el particionado del disco duro, es sabido que al menos debemos crear dos particiones obligatorias, el área de intercambio y la partición raíz representada por (/) pero si disponemos de suficiente espacio podemos crear una partición para /home para guardar los datos de los usuarios en caso de catástrofe, claro que también podemos crear una partición para /var y otra para /usr.

NOTA: Algo muy importante a tener en cuenta en este punto es que si vamos a brindar servicios de ftp y/o http debemos crear una partición para /srv con suficiente espacio ya que es en este punto donde se alojará el ftp y los sitios web.

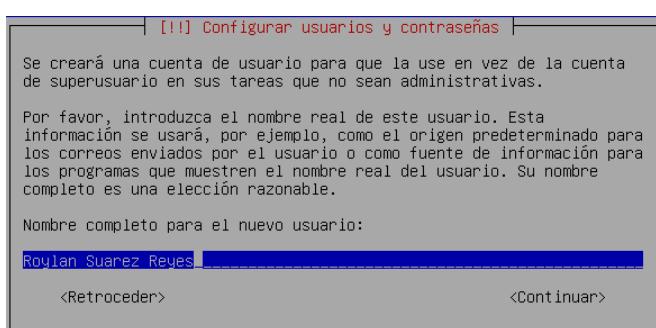
La Distro



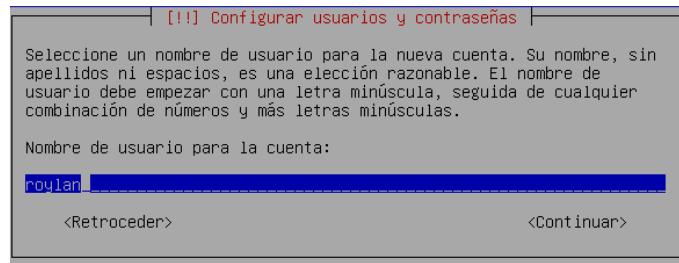
En el próximo paso nos aseguramos de que el particionado es correcto confirmamos, una vez aceptado todos los datos de la partición o disco duro no se podrán recuperar



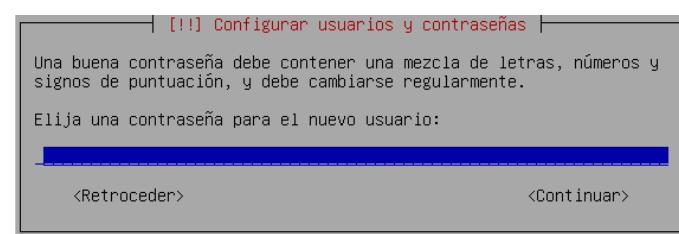
Finalizado el particionado del disco duro se creará una cuenta de usuario para usarla en vez de la cuenta de superusuario. Escribimos el nombre completo de nuevo usuario como lo muestra la figura y presionamos Enter



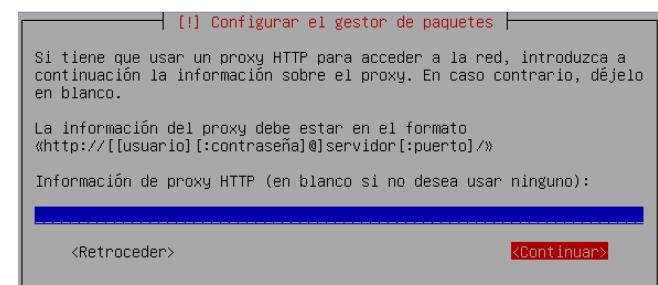
Seleccionamos un nombre de usuario para iniciar sesión, es aconsejable escribirlo en minúscula y sin espacios.



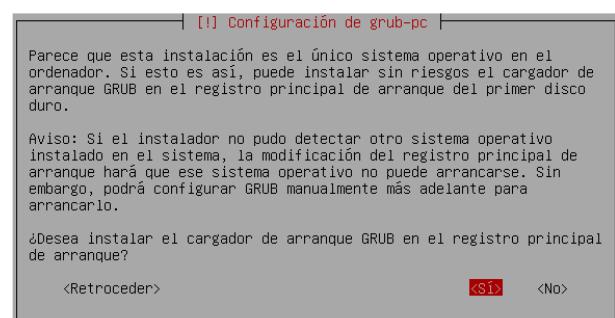
Debemos igualmente establecer una buena contraseña utilizando una combinación de letras, números y signos de puntuación, recuerde de que esta será la llave de acceso al servidor.



Si se encuentra detrás de un proxy cache como squid, introduzca la información sobre el proxy, en caso contrario déjelo en blanco.

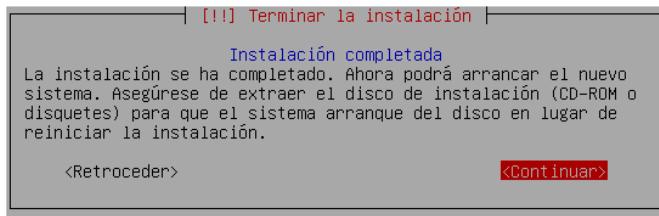


Una vez suministrada toda la información comenzará el proceso de instalación, el tiempo estará en dependencia de las condiciones del ordenador. Al terminar nos preguntará si deseamos instalar el gestor de arranque GRUB.

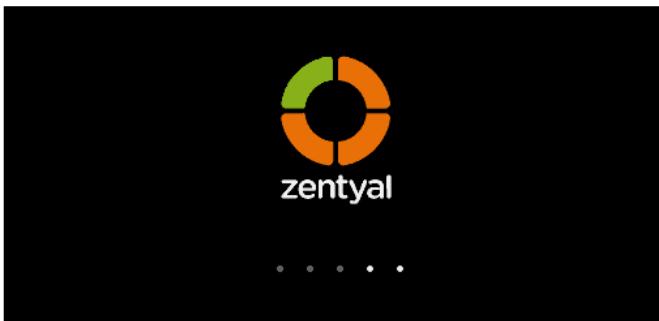


Cuando se complete la instalación nos aseguraremos de extraer el CD-ROM para que el nuevo sistema arranque desde el disco duro.

La Distro



Tras el primer reinicio podemos ver el splash del sistema



En todo el proceso anterior hemos instalado el sistema base de Zentyal, ahora nos resta instalar los módulos de los servicios que brindaremos en nuestro servidor.

El sistema arrancará una interfaz gráfica con un navegador que permite acceder a la interfaz de administración, y aunque tras este primer reinicio el sistema haya iniciado el entorno gráfico automáticamente, de aquí en adelante, necesitará autenticarse antes de que éste arranque



Una vez autenticado por primera vez en la interfaz web comienza un asistente de configuración, en primer lugar podremos seleccionar qué funcionalidades queremos incluir en nuestro sistema. Existen dos métodos para esta selección:

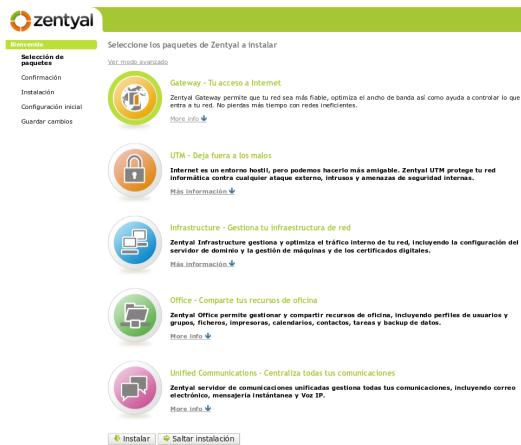
Simple:

Se seleccionará el o los perfiles de instalación que deseemos para nuestro servidor. Un perfil de instalación es un conjunto de paquetes que agrupan una serie de funcionalidades según la tarea que vaya a desempeñar el servidor.

Avanzado:

Se seleccionarán los paquetes de manera individual y sus dependencias se resolverán automáticamente al confirmar la instalación.

La selección simple se realiza a través de la lista de perfiles de instalación disponibles. Estos perfiles de instalación facilitan y simplifican la realización de despliegues de Zentyal en los escenarios más típicos, aunque esto no impide combinar la funcionalidad disponible conforme las necesidades lo requieran.



Perfiles de Zentyal que podemos instalar:

Zentyal Gateway:

Zentyal actúa como la puerta de enlace de la red local ofreciendo un acceso a Internet seguro y controlado.

Zentyal Unified Threat Manager:

Zentyal protege la red local contra ataques externos, intrusiones, amenazas a la seguridad interna y posibilita la interconexión segura entre redes locales a través de Internet u otra red externa.

La Distro

Zentyal Infrastructure:

Zentyal gestiona la infraestructura de la red local con los servicios básicos: DHCP, DNS, NTP, servidor HTTP, etc.

Zentyal Office:

Zentyal actúa como servidor de recursos compartidos de la red local: ficheros, impresoras, calendarios, contactos, perfiles de usuarios y grupos, etc.

Zentyal Unified Communications:

Zentyal se convierte en el centro de comunicaciones de la empresa, incluyendo correo, mensajería instantánea y Voz IP.

Podemos seleccionar varios perfiles para hacer que Zentyal tenga, de forma simultánea, diferentes roles en la red.

Sin embargo, mediante la selección avanzada aparecerá la lista de módulos de Zentyal y se podrán seleccionar individualmente aquellos que se necesiten.

The screenshot shows the 'Package Selection' step of the Zentyal setup wizard. It lists various software components with their latest versions and checkboxes for selection. Components include Antivirus, Application Protocols, Backup, Certification Authority, DHCP Service, DNS Service, FTP, File Sharing, Firewall, Groupware, Groupware (Zarafa), HTTP Proxy (Cache and Content Filter), Intrusion Detection System, Jabber (Instant Messaging), Mail Filter, Mail Service, Monitor, NTP Service, and Network Configuration. The 'Select' column contains checkboxes for each item.

Para nuestro ejemplo usaremos una instalación del perfil de Gateway únicamente.

Al terminar la selección, se instalarán también los paquetes adicionales necesarios y además si hay algún complemento recomendado se preguntará si lo queremos instalar también. Esta selección no es definitiva, ya que posteriormente podremos instalar y desinstalar el resto de módulos de Zentyal a través de la gestión de software.

This screenshot shows a confirmation dialog box. It lists the packages to be installed: Certification Authority, Network Objects, Network Configuration, Users and Groups, Traffic Shaping, Monitor, VPN Service, and Gateway Suite. To the right, a list of recommended packages is shown: Backup, Network Services, Firewall, HTTP Proxy (Cache and Content Filter), Layer-7 Filter, RADIUS, and Zentyal Cloud Client. At the bottom are 'Aceptar' and 'Cancelar' buttons.

Quizá desee instalar también los siguientes paquetes recomendados:

Filtrado de nivel 7 para el Moldeado de Tráfico (ebox-i7-protocols)

Zentyal created by eBox Technologies S.L.

El sistema comenzará con el proceso de instalación de los módulos requeridos, mostrando una barra de progreso donde además podemos leer una breve introducción sobre las funcionalidades y servicios adicionales disponibles en Zentyal.

Instalando

Bienvenido!

• ¡Gracias por haber elegido Zentyal!
• Estás usando la versión 2.0 de Zentyal, el servidor Linux para pequeña y mediana empresa.
• Esperamos que Zentyal te parezca fácil y útil! Sin embargo, si se encuentra con problemas, irogamos consulta en el Foro de Zentyal!
• Mientras se instalan los paquetes de Zentyal que haya elegido, esta presentación le enseñará qué más ofrecemos además del software de servidor Zentyal!
<http://forum.zentyal.org/>



Instalando paquetes

Operación actual: down0 Unpacking openssl-blacklist (from .../openssl-blacklist_0.5-2_all.deb) ...

1%

1 of 170 operaciones realizadas

Una vez terminado el proceso de instalación el asistente configurará los nuevos módulos realizando algunas preguntas.

En primer lugar se solicitará información sobre la configuración de red, definiendo para cada interfaz de red si es interna o externa, es decir, si va a ser utilizada para conectarse a Internet u otras redes externas o si está conectada a la red local. Se aplicarán políticas estrictas en el cortafuegos para todo el tráfico entrante a través de interfaces de red externas.

Wizard de configuración inicial

Saltar Siguiente

Interfaces de Red



Selecciona interfaces externas de red

No será posible conectar a la interfaz de administración de Zentyal a través de ellas

eth0: Internal External

Saltar Siguiente

La Distro

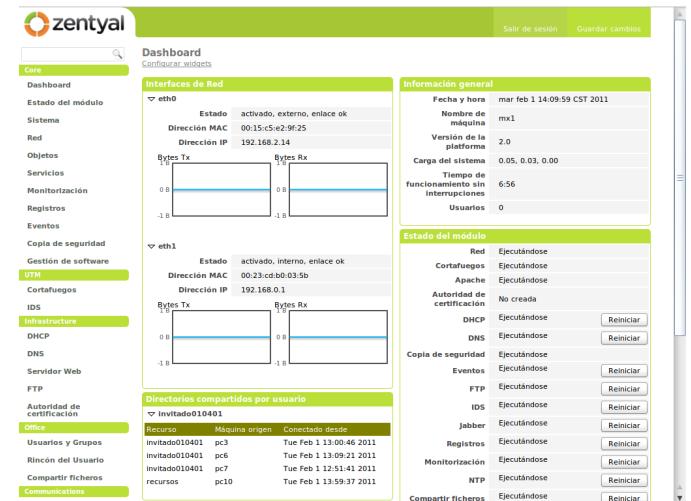


A continuación tendremos que seleccionar el tipo de servidor para el modo de operación del módulo Usuarios y Grupos. Si sólo vamos a tener un servidor elegiremos- Servidor stand-alone. Si por el contrario estamos desplegando una infraestructura maestro-esclavo con varios servidores Zentyal y gestión de usuarios y grupos centralizada o si queremos sincronizar los usuarios con un Microsoft Active Directory,



elegiremos-Configuración avanzada.

nes, se procederá a la configuración de cada uno de los módulos instalados.



Cuando finalice el proceso de guardar cambios ya podremos acceder al Dashboard: ¡nuestro servidor Zentyal ya está listo!

Hasta aquí este artículo de demuestra el proceso de instalación de Zentyal, en el próximo número de Linvix, veremos como ajustar el resto de los parámetros de la red y configurar un servidor DNS.



Lic. Roylán Suárez Reyes
linvix@gmail.com
linvix.wordpress.com

Una vez hayan sido respondidas estas cuestio-

Servidores

Instalación Servidor OpenVPN y cliente Windows

En breve OpenVpn es una aplicación de código abierto SSL VPN, que trabaja sobre un modelo cliente/servidor , con un servidor ejecutándose en nuestro host y clientes conectándose al servidor y creando túneles VPN.



Instalamos el software openvpn

```
sudo aptitude install openvpn
```

Lo agregamos para que se cargue al inicio:

```
sudo update-rc.d openvpn defaults
```

Editamos el archivo de configuración:

```
/etc/openvpn/server.conf
```

```
# Configuración de la red
```

```
dev tun  
port 1194  
proto udp  
server 10.8.0.0 255.255.255.0  
keepalive 10 120
```

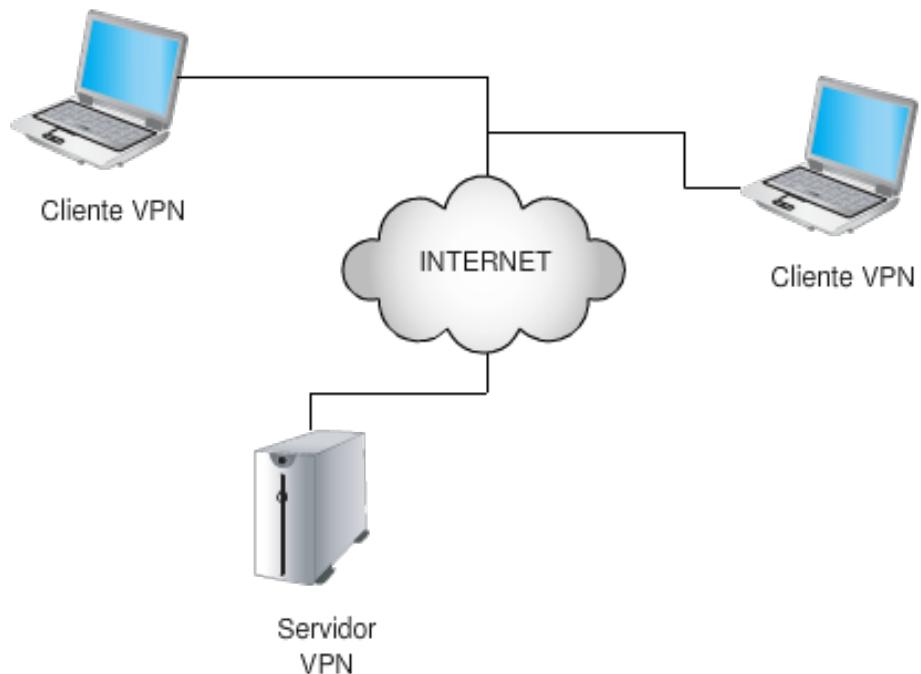
```
# Configuración del registro
```

```
log-append /var/log/openvpn.log  
status /var/log/openvpn-status.log  
verb 4  
mute 20
```

```
# Configuración de la seguridad
```

```
user nobody  
group nobody  
persist-key  
persist-tun
```

```
# Compresión  
comp-lzo
```



Servidores

Explicaré brevemente cada una de las opciones:

dev tun: Dispositivo virtual que será el túnel vpn, este se creará como tun0.

port : Puerto de conexión.

proto : protocolo.

server: dirección ip del servidor y direcciones ip para nuestros clientes. Por defecto nuestro servidor tomará la IP 10.8.0.1 y asignará las direcciones restantes a los clientes.

Keepalive : el valor 10 , le dice a vpn que haga un ping cada 10 segundos a la conexión y el valor 120 es el tiempo en segundos que esperará una respuesta openVPN , si no recibe la respuesta nos notificará.

Persist-key y persist-tun: están relacionados con la seguridad, permiten a openvpn tener los privilegios para trabajar con las interfaces de red y certificados ssl.

Para nuestros túneles vpn, podemos crear y firmar nuestros propios certificados utilizando una CA (autoridad certificadora).

Vamos a crear un certificado de servidor para el servidor vpn.

```
sudo openssl req -new -newkey rsa:2048 -nodes -keyout serv_vpn.com.ar.key -out serv_vpn.com.ar.req
```

Generating a 2048 bit RSA private key

.....+++

.....+++

writing new private key to 'serv_vpn.com.ar.key'

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:AR

State or Province Name (full name) [Some-State]:Cordoba

Locality Name (eg, city) []:Cordoba

Organization Name (eg, company) [Internet Widgits Pty Ltd]:CBA

Organizational Unit Name (eg, section) []:CBA

Common Name (eg, YOUR name) []:resr

Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request

A challenge password []:

An optional company name []:

Si no tenemos una CA vamos a crearla, haciendo los siguientes pasos:

Servidores

```
sudo mkdir -p /etc/CA  
cd /etc/CA/  
sudo mkdir private  
sudo mkdir newcerts  
echo "01" | sudo tee /etc/CA/serial  
sudo touch /etc/CA/index.txt
```

Copiamos el archivo siguiente a nuestro directorio CA:

```
sudo cp /etc/ssl/openssl.cnf /etc/CA/
```

Editamos el archivo openssl.cnf y buscamos la directiva [CA_default] y debajo habrá una opción llamada dir la editaremos y tendrá que quedar así:

dir=.

Ejecutamos lo siguiente para crear el certificado y la clave privada para nuestra CA:

```
sudo openssl req -new -x509 -newkey rsa:4096 -keyout private/cakey.pem -out cacert.pem
```

Después que ejecutamos este comando nos pedirá una password en un item como el siguiente:

Enter PEM pass phrase

La password que pongamos acá nos servirá en el paso siguiente para firmar nuestro certificado.

Ahora firmaremos nuestro certificado vpn con la CA creada.

```
sudo openssl ca -out serv_vpn.com.ar.cert -config ./openssl.cnf -infiles serv_vpn.req
```

Using configuration from ./openssl.cnf

Enter pass phrase for ./private/cakey.pem: # password anterior del paso 6

Check that the request matches the signature

Signature ok

Certificate Details:

Serial Number: 1 (0x1)

Validity

Not Before: Nov 23 15:11:53 2010 GMT

Not After : Nov 23 15:11:53 2011 GMT

Subject:

countryName = AR

stateOrProvinceName = Cordoba

organizationName = test

organizationalUnitName = test

commonName = rxxxx

X509v3 extensions:

Servidores

```
X509v3 Basic Constraints:  
    CA:FALSE  
Netscape Comment:  
    OpenSSL Generated Certificate  
X509v3 Subject Key Identifier:  
    48:B4:05:CA:32:29:55:16:A1:A9:5B:FF:A9:27:C9:35:15:EA:4C:F7  
X509v3 Authority Key Identifier:  
    keyid:87:2D:F6:0C:56:55:BB:77:83:01:11:61:CC:2E:47:6F:E7:8A:B0:25
```

Certificate is to be certified until Nov 23 15:11:53 2011 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]
Write out database with 1 new entries
Data Base Updated

Para potenciar nuestra seguridad vamos a crear el “parametro Diffie-Hellman”

```
sudo openssl dhparam -out ./dh2048.pem 2048
```

```
Generating DH parameters, 2048 bit long safe prime, generator 2  
This is going to take a long time
```

```
.....+.....+.....  
.....+.....+.....  
.....+.....+.....  
.....+.....+.....  
.....++*++*
```

Aseguramos nuestros archivos:

```
sudo chmod 0600 serv_vpn.com.ar.key  
sudo chown root:root serv_vpn.com.ar.*
```

Con todo lo anterior nuestro server.conf quedará de la siguiente manera:

#Configuración de la red

```
dev tun  
port 1194  
proto udp  
server 10.8.0.0 255.255.255.0  
keepalive 10 120
```

#Configuración del registro

```
log-append /var/log/openvpn.log
```

Servidores

```
status /var/log/openvpn-status.log
verb 4
mute 20

#configuración del certificado

ca /etc/CA/cacert.pem
dh /etc/openvpn/dh2048.pem
cert /etc/openvpn/serv_vpn.com.ar.cert
key /etc/openvpn/serv_vpn.com.ar.key
```

Configuración de la seguridad

```
user nobody
group nobody
persist-key
persist-tun
```

#Compresión

comp-lzo

Con esto ya tendremos nuestro servidor openvpn ahora a configurar las sucursales:

Configuración de las sucursales

Generamos un certificado para nuestro servidor de sucursal (serv_hp)

```
cd /etc/CA
```

```
sudo openssl ca -out serv_hp.cert -config ./openssl.cnf -infiles /etc/openvpn/serv_hp.com.ar.req
```

Using configuration from ./openssl.cnf

Enter pass phrase for ./private/cakey.pem: #password del apartado anterior paso 6

Check that the request matches the signature

Signature ok

Certificate Details:

Serial Number: 2 (0x2)

Validity

Not Before: Nov 23 15:24:55 2010 GMT

Not After : Nov 23 15:24:55 2011 GMT

Subject:

countryName = AR

stateOrProvinceName = Cordoba

organizationName = xx

organizationalUnitName = xxx

commonName = xxx

X509v3 extensions:

X509v3 Basic Constraints:

CA:FALSE

Netscape Comment:

Servidores

```
OpenSSL Generated Certificate
X509v3 Subject Key Identifier:
54:B6:3B:F0:4E:60:42:08:7B:11:8F:8B:E0:28:69:10:BA:88:0E:33
X509v3 Authority Key Identifier:
keyid:87:2D:F6:0C:56:55:BB:77:83:01:11:61:CC:2E:47:6F:E7:8A:B0:25
```

Certificate is to be certified until Nov 23 15:24:55 2011 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated

Pasamos los siguientes archivos al servidor de sucursal, en mi caso lo pase por ftp:

```
sftp user@serv
Connecting to serv ...
sftp> put serv_hp.com.ar.cert
sftp> put serv_hp.com.ar.key
sftp> put cacert.pem
```

Como nuestro servidor de sucursal es un Windows, tendremos que instalar el soft openvpn cliente que se baja desde acá <http://openvpn.net/>:

Lo instalaremos, no requiere ninguna complejidad, cuando finalice nos iremos al directorio:

C:\Archivos de programa\OpenVPN\config

Crearemos un archivo llamado: cliente1-udp-1194.ovpn que contendrá lo siguiente:

```
tls-client
dev tun
client
proto udp
remote servidorVPN-o-IP.com 1194
```

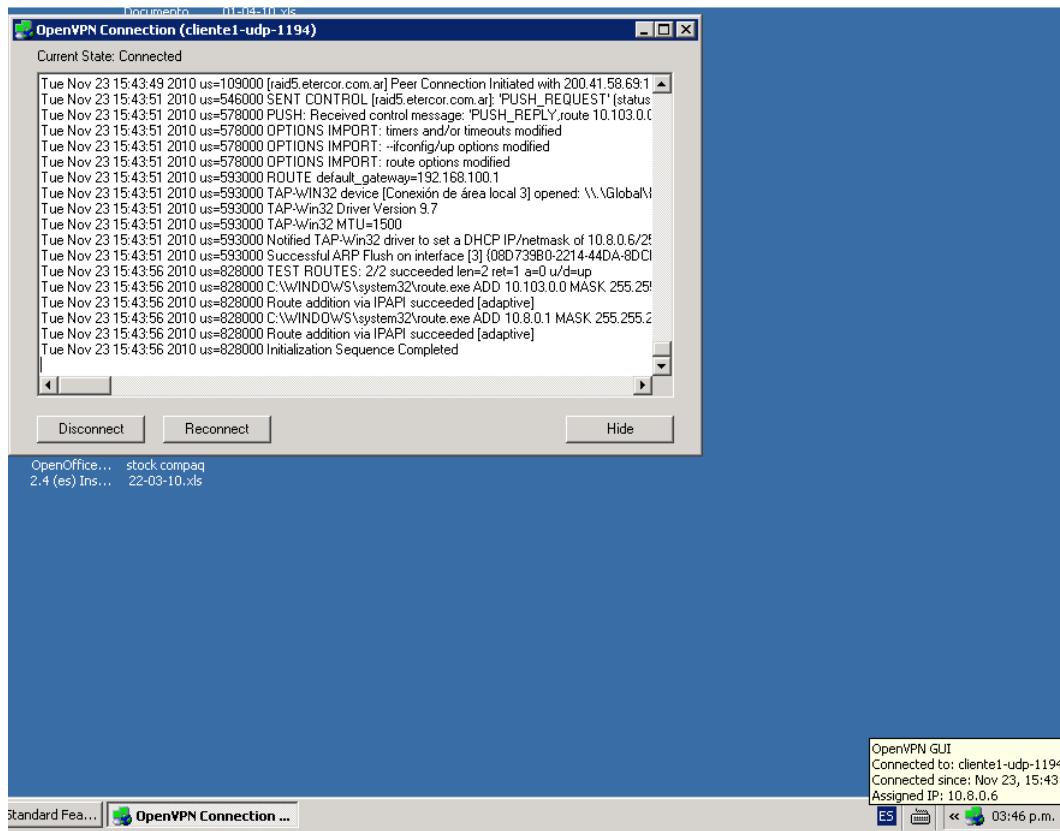
```
#----- SECCION DE LLAVES -----
ca "c:\\Archivos de programa\\OpenVPN\\config\\cacert.pem"
cert "c:\\Archivos de programa\\OpenVPN\\config\\serv_hp.com.ar.cert"
key "c:\\Archivos de programa\\OpenVPN\\config\\serv_hp.com.ar.key"
#
#-----
comp-lzo
verb 5
```

Cabe aclarar que los archivos que pasamos anteriormente por ftp los tenemos que descargar en ese directorio: C:\Archivos de programa\OpenVPN\config
En la sección remote debemos poner un servidor que sea accesible por su nombre o por su IP

Servidores

a través de Internet junto al puerto.

Ahora iniciaremos nuestro CLIENTE vpn desde el icono:



si vemos la IP observaremos:

Adaptador Ethernet Conexión de área local 3 :

Sufijo de conexión específica DNS:

Dirección IP.....: 10.8.0.6

Máscara de subred.....: 255.255.255.252

Puerta de enlace predeterminada : #

¿Qué es la IP que nos dio nuestro servidor vpn?.

Ahora hacemos ping a nuestro servidor vpn

C:\Documents and Settings\Administrador>ping 10.8.0.1

Haciendo ping a 10.8.0.1 con 32 bytes de datos:

Respuesta desde 10.8.0.1: bytes=32 tiempo=36ms TTL=64

Respuesta desde 10.8.0.1: bytes=32 tiempo=43ms TTL=64

Respuesta desde 10.8.0.1: bytes=32 tiempo=42ms TTL=64

Estadísticas de ping para 10.8.0.1:

Paquetes: enviados = 3, recibidos = 3, perdidos = 0

Servidores

(0% perdidos),

Ahora desde el lado del servidor vpn:

ifconfig:

```
eth0      Link encap:Ethernet HWaddr a2:00:54:3a:b3:09  
          inet addr:10.103.0.5 Bcast:10.103.0.255 Mask:255.255.255.0  
          inet6 addr: fe80::a000:54ff:fe3a:b309/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
          RX packets:5987 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:2113 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:694959 (694.9 KB) TX bytes:675703 (675.7 KB)  
          Interrupt:21 Base address:0xec00  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
          UP LOOPBACK RUNNING MTU:16436 Metric:1  
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:0  
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)  
  
tun0     Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  
          inet addr:10.8.0.1 P-t-P:10.8.0.2 Mask:255.255.255.255  
          UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1  
          RX packets:14 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:100  
          RX bytes:840 (840.0 B) TX bytes:840 (840.0 B)
```

Ejecutamos un ping al cliente vpn:

```
ping 10.8.0.6
```

```
PING 10.8.0.6 (10.8.0.6) 56(84) bytes of data.  
64 bytes from 10.8.0.6: icmp_seq=1 ttl=128 time=38.2 ms  
64 bytes from 10.8.0.6: icmp_seq=2 ttl=128 time=36.8 ms  
64 bytes from 10.8.0.6: icmp_seq=3 ttl=128 time=34.8 ms  
64 bytes from 10.8.0.6: icmp_seq=4 ttl=128 time=34.8 ms
```

Eso es todo por ahora, cualquier consulta no duden en enviarme un Email. Saludos desde Argentina.

Ing. Roque Moyano
Administrador de servidores linux
Argentina
roque.ing@gmail.com

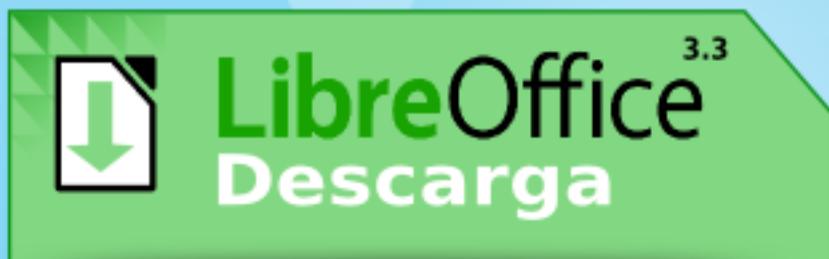
Liberate, usa:



LibreOffice

The Document Foundation

LibreOffice está abierta a cualquier persona que esté de acuerdo con sus valores fundamentales y contribuya a sus actividades.



www.libreoffice.org

Programación

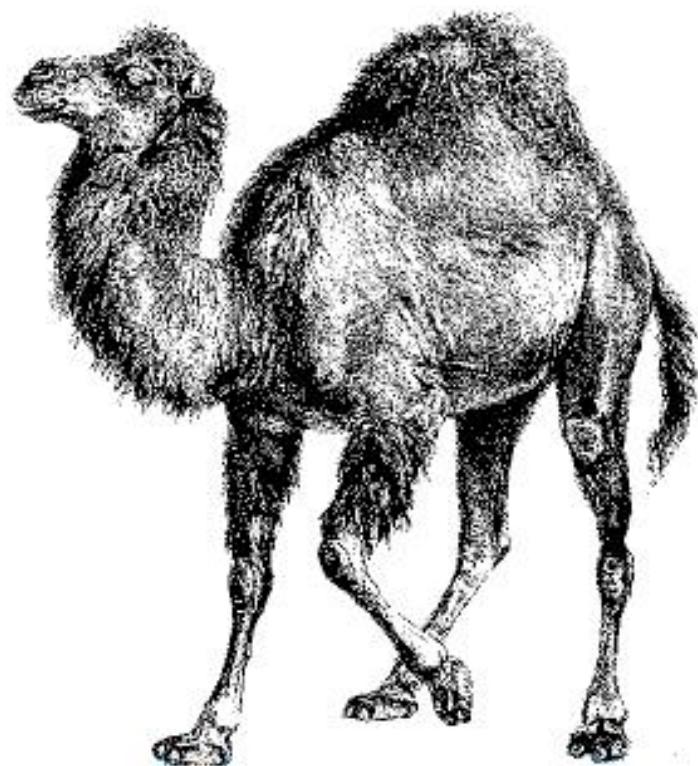
Introducción a Perl

Carlos Salguero
cfsalguero@gmail.com

La intención de este artículo es comenzar una introducción al lenguaje Perl, uno de los más populares dentro del ambiente Linux.

Como programador que soy, voy a hacer una brevíssima reseña y después nos pondremos directamente a ver código por dos razones: Por un lado, esto no pretende ser una iniciación a la programación sino una iniciación al lenguaje Perl para personas que al menos tienen una idea básica de programación y en segundo término, estoy convencido de que la única forma de aprender a programar es programando.

Perl es un lenguaje creado por Larry Wall que tiene muchas cosas en común con C, comandos de Unix/Linux y algunos otros lenguajes. En principio puede ser un lenguaje un tanto confuso pero la realidad es que es muy simple de leer un programa escrito en Perl. Por supuesto, la legibilidad de un programa no depende de un lenguaje en particular sino de la habilidad del programador de escribir programas claros; uno puede escribir hermosos programas en Perl o C o Java o puede escribir programas Perl, C o Java imposibles de leer. Hablando particularmente de Perl que es de lo que trata esta nota, les recomiendo que traten siempre de escribir programas claros, porque Perl es un lenguaje tan flexible que les va a dejar escribir casi cualquier cosa y lo va a entender, los que no van a entender son ustedes mismos cuando retomen el programa después de cierto tiempo y no recuerden el por qué hicieron cada cosa. Nunca menosprecien el valor de un “enter” bien puesto; ocupa solo un carácter pero la claridad que puede aportar es realmente significativa.



¿Para qué se usa Perl? Perl muchas veces es encasillado como un lenguaje para procesar grandes volúmenes de texto. En cierta forma eso es correcto porque ahí radica la mayor fortaleza del lenguaje, en el procesamiento de texto, cosa que se logra fundamentalmente gracias a su potentísimo motor de expresiones regulares. Tanto es así que muchas implementaciones de expresiones regulares se comparan o tratan de llegar a ser “compatibles con Perl”. Para los que no sepan qué es una expresión regular les dejo una breve explicación sobre la que iremos avanzando en los próximos artículos: una expresión regular es una forma de escribir una “fórmula” para buscar determinados patrones dentro de un texto. Por ejemplo /linux\d+/ busca dentro de un texto la palabra linux seguida de uno o más dígitos numéricos.

Volviendo a la pregunta ¿Para qué se usa Perl?, la respuesta es simple: para todo. Perl es capaz de procesar texto, hacer páginas

Programación

Web como php, conectarse a base de datos, abrir puertos de comunicaciones, interactuar con el sistema operativo, etc, etc, etc. Una de las mayores fortalezas de Perl es realmente su amplia biblioteca de librerías para hacer cualquier cosa que uno pueda imaginar y a la que puede acceder como curiosidad entrando a <http://www.cpan.org>.

Manos a la obra

Para respetar los estándares internacionales sobre cursos de programación, vamos a empezar con el programa más usado en la historia de la programación, gentileza de Kernighan & Ritchie: Hello World!!!

```
#!/usr/bin/perl -w  
print ("Hello, world!\n");
```

La primer línea es como en casi todo script de Linux, donde le indicamos al sistema operativo como debe ejecutar nuestro programa. Seguramente ya lo han visto en algún script del sistema operativo donde pueden forzar el uso de shell o bash. Aquí le indicamos que nuestro interprete/compilador de perl está en /usr/bin/perl y además le pasamos un parámetro -w, que le indica a Perl que nos muestre los mensajes de advertencia (warnings).

Si no saben donde esta instalado Perl en su distribución en la línea de comandos pueden escribir:

```
which perl
```

y les dirá donde está el binario.

La segunda línea es un simple comando print que nos mostrará el mensaje que pongamos entre comillas, en nuestro caso, Hello, world! seguido de un salto de línea (\n). Noten que todas las líneas de programas perl terminan con un punto y coma, excepto las líneas que abren o cierran un bloque como ser un bloque if o while.

Lo más común es tener scripts en archivos de

texto con extensión .pl. En realidad la extensión no es necesaria en Linux, pero se usa por convención para identificar rápidamente de que tipo de archivo se trata. De ese modo los .pl son scripts de perl y los .sh son scripts de shell.

Si han escrito este programa en su editor de textos favorito (vi en mi caso) y lo han grabado con un nombre como ser hello.pl pueden ejecutarlo desde la línea de comandos, pero antes debemos asignarle permisos de ejecución haciendo

```
chmod u+wx hello.pl
```

Una vez que nuestro programa tiene permisos, podremos probarlo haciendo:

```
./hello.pl
```

con lo que veremos un hermoso mensaje Hello, world!

Variables en Perl

Perl tiene algunos tipos básicos de variables, que son los escalares (variable comunes e independientes), arrays, hashes y referencias. Los escalares comienzan siempre con un signo \$, los arrays con @ y los hashes con % y las referencias (los apuntadores de C) al tener solo una dirección de memoria, son escalares por que también comienzan con un signo \$.

Más adelante iremos viendo las diferencias y utilidades de cada tipo, pero por ahora nos centraremos en los escalares.

Como mencione antes las variables escalares comienzan con un signo \$ y no tienen un tipo específico. Por ejemplo, podemos escribir estas líneas una debajo de la otra:

```
$a = 10;  
$a = "hola mundo";  
$a = "probando ".(10+3);
```

y Perl asignará un número 10 a la variable a,

Programación

Luego le asignará un texto y luego un texto que es el resultado de concatenar un texto con una operación matemática sin ningún tipo de problemas.

También podemos limpiar una variable asignándole el equivalente al null de otros lenguajes que en Perl es undef: \$a=undef;

Ingresando datos desde el teclado:

Vemos el siguiente programa:

```
1. #!/usr/bin/perl -w
2. print "Cuál es su nombre? ";
3. $name = <STDIN>;
4. chomp ($name);
5. print "Hola, $name!\n";
```

Analicemos línea por línea:

1. Como ya vimos, es donde indicamos el lugar donde se encuentra instalado Perl.
2. Imprimimos un mensaje invitando a la persona a ingresar su nombre.
3. Cuando la persona ingreso su nombre en la línea 2, pulso enter para confirmar la entrada. La instrucción chomp saca el enter al final de una cadena de textos. Si la cadena no tiene un enter final, chomp no hará nada.
4. Imprimimos un mensaje saludando a la persona por su nombre.

Creo que la línea mas extraña hasta el momento es la línea 3 donde asignamos a la variable \$name el valor de <STDIN>.

En Perl, la forma de ingresar algo por teclado es abriendo un archivo que en este caso es la entrada estandard (teclado) STDIN.

<....> es la forma de decirle a Perl que lea datos desde un archivo.

En futuras entregas veremos en más detalles cómo leer otros archivos reales que no sean la entrada standard.

How-To

CUPS. Instalar una impresora Epson LX 300+ en Ubuntu

En este artículo vamos a ver como instalar una impresora Epson LX 300+ en linux, pues esta es una de las impresoras que más abundan en nuestro país y hasta el momento no hemos encontrado driver para ella, esta guía ayudará a que el proceso de migración sea menos traumático.

Utilizando CUPS podemos instalar otros modelos y lograr simular una impresora Epson LX300+ y que la impresión salga con la calidad requerida, pero para ello primero veremos que es CUPS y como se instala.

(CUPS) no es más que las siglas de Common Unix Printing System (Sistema de impresión común de Unix) es un sistema de impresión modular para sistemas operativos de tipo Unix que permite que un computador actúe como servidores de impresión. Un computador que ejecuta CUPS actúa como un servidor que puede aceptar tareas de impresión desde otros computadores cliente, los procesa y los envía al servidor de impresión apropiado.

Primero abrimos el navegador y escribimos la siguiente dirección: <http://localhost:631> pues en las últimas versiones de Ubuntu viene por defecto.

Si no está instalado, abrimos una consola y escribimos:

```
#sudo aptitude install cups cupsd cupsys cups-common
```

Recuerden que tenemos que conectar la impresora a la PC.



Ahora nos vamos al navegador nuevamente y escribimos la dirección anterior obteniendo una salida como esta:

CUPS 1.4.3

CUPS es el sistema de impresión de código abierto basado en estándares desarrollado por Apple Inc. para Mac OS® X y otros sistemas operativos tipo UNIX®.

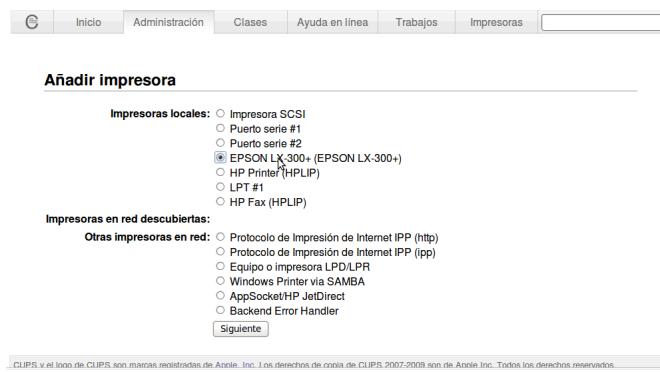
CUPS para usuarios	CUPS para administradores	CUPS para desarrolladores
Descripción de CUPS	Añadiendo impresoras y clases	Introducción a la programación de CUPS
Impresión desde la línea de comandos y opciones	Gestionando políticas de funcionamiento	La API de CUPS
Qué hay de nuevo en CUPS 1.4	Contabilidad básica de impresora	Programación de filtros y programas de conexión
Foro de usuarios	Seguridad del servidor	Las APIs HTTP e IPP
	Usando autenticación Kerberos	La API PPD
	Usando impresoras de red	La API Raster
		Referencia del archivo de información del sistema

How-To

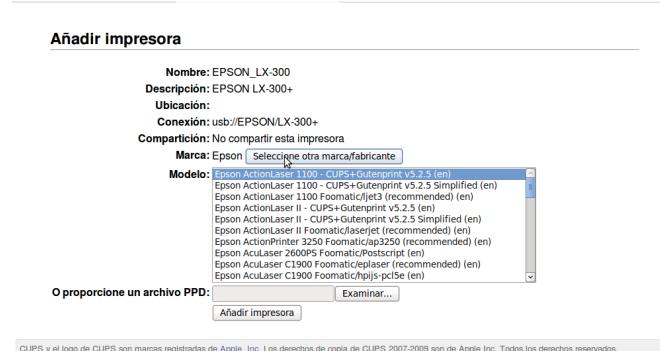
Ahora damos click en la pestaña Administración :



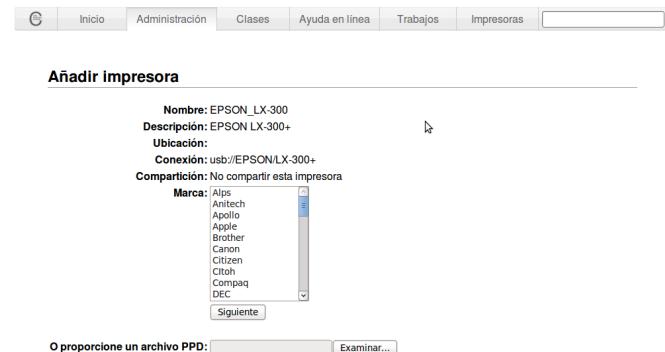
Para instalar la impresora damos clic en la pestaña Añadir impresora y nos pedirá un usuario y contraseña, debemos introducir la de nuestra PC, o mejor dicho una cuenta con privilegios de administración:



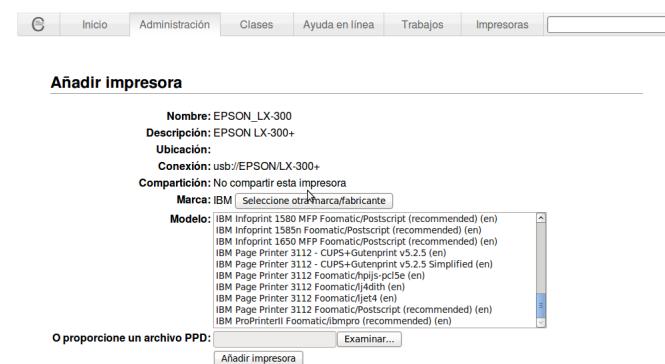
Aquí marcamos la impresora que tenemos conectada que aparece en la lista que nos muestra, y damos siguiente:



Pero como no tenemos los drivers de la Epson LX300+ entonces debemos seleccionar otra marca de impresora, que como había dicho antes funciona bien con esta impresora, así que damos click en el botón que dice Seleccionar Marca/Fabricante.



Luego buscamos en la lista que se muestra la marca IBM y damos siguiente.



Ahora buscamos el modelo (IBM ProPrinter II Foomatic/ibmpro (recommended)(en)) y damos clic en el botón Añadir Impresora y en unos segundos quedará instalada la impresora.

Osniel Cabeza García
linvix@gmail.com

How-To

Cómo regular el envío y recepción de correos electrónicos a través de conexiones PPP.

Lic. Michel Vega Fuenzalida
michelvf@ijv.sld.cu

Aunque en nuestros tiempos ya el modem está pasando al pasado, todavía en Cuba se usa y en alto grado, para quienes aún tienen conexión a través de PPP y por esa vía gestionan el correo, va este trabajo.

Por defecto en las distribuciones de GNU/Linux®, no tienen la posibilidad de controlar en envío y recepción de correos electrónicos a través de conexiones PPP, usando los paquetes fetchmail y sendmail o postfix, este script es un intento para lograr este objetivo.

Problema

La conexión con nuestro ISP (siglas en Inglés, en español: Proveedor de Servicios de Internet) es a través de enlaces conmutados, es decir, línea telefónica y modem, el contrato con Enet es de una cuenta Multipop, donde se nos dan 40 horas al mes a \$40.00 CUC. Por cada hora pasada de las 40, se cobrará \$4.00 CUC, por lo que se debe garantizar que con las horas asignadas se logre enviar y recibir todos los mensajes del mes. Como vemos el recurso que se debe vigilar es el tiempo de conexión. No olvidemos que con una cuenta Multipop se pueden crear tantos usuarios como se deseen, provocando esto que aumente el tráfico de correo por cada usuario que se añada.

En distribuciones GNU/Linux®, no existe un software, que logre regular el envío y recepción de mensajes a través de conexiones PPP, a no ser que se regule por medio del Cron, pero esta solución no es muy eficiente.

En nuestro Servidor, como la conexión a nuestro ISP es a través de conexiones PPP, usamos los software fetchmail y sendmail o postfix como MTA. Fetchmail para descargar los mensajes de nuestro ISP a través del protocolo POP3 y entregarlos a través de SMTP al MTA.

Por lo general, los agentes de correos de usuario, al gestionar la mensajería, primero envían lo pendiente y después reciben. Esta es la idea que seguimos al hacer el script para los MTA.

How-To

Solución

```
fetchmail -q
ifup ppp0
## Si usamos sendmail descomentar la línea siguiente
#sendmail -q
## Si usamos postfix descomentar la línea siguiente
#postfix flush
a=0; b=0;
until (( "$a" == "1" )); do
## Si usamos sendmail descomentar la línea siguiente
# verif=`mailq | grep -ic "Total requests: 0"`
## Si usamos postfix descomentar la línea siguiente
# verif=`mailq | grep -ic "Mail queue is empty"`
if [ "$verif" -eq 1 ]; then a=1;
else sleep 10;
fi
done
fetchmail -d
until (( "$b" == "1" )); do
fetch=`tail -1 /var/log/fetchmail.log | grep -ic "fetchmail: sleeping at"`
if [[ $fetch > "0" ]]; then b=1;
fi
done
fetchmail -q
ifdown ppp0
exit 0
```

Este script se salva en un fichero nombrado... digamos: regular-correo.sh, en /root/, le damos permisos de ejecución y que solo root sea el dueño:

```
# chmod 700 /root/ regular-correo.sh
```

Lo que falta es regularlo por el cron. Editar el fichero /etc/crontab con

```
# crontab -e
```

y agregar la siguiente línea:

```
*/20 8-17 * * * root /root./regular-correo.sh
```

Esto hace que el fichero se ejecute cada 20 minutos.
Para saber más sobre el crontab: man 5 crontab.

How-To

Cómo cifrar y utilizar el directorio Private en Ubuntu

Paco Revilla
www.espaciolinux.com

A partir de Ubuntu 8.10 se puede crear un directorio cifrado en nuestro carpeta personal bajo el nombre Private, en donde podemos incluir información importante que no podrá ser accesible de manera externa sin conocer la clave que le hayamos asignado.

Algunos asuntos a considerar:

- El directorio cifrado se puede utilizar normalmente, cuando iniciemos sesión en nuestra cuenta. A pesar de que podamos ver los datos, desde afuera de nuestra cuenta no pueden hacerlo.
- Los datos no son accesibles, por ejemplo, para el superusuario del sistema u otros usuarios con privilegios, en modo de rescate o a través de un LiveCD o herramienta de rescate.
- A pesar que los datos se mantienen cifrados, pueden ser borrados, contando con los privilegios adecuados.
- Es importante no olvidar la clave de cifrado para en caso de tener que recuperar manualmente los datos, podamos hacerlo.
- Podemos utilizar este directorio en portátiles para mantener seguros nuestros datos. Existen otras formas más seguras y completas de hacerlo, por ejemplo, cifrando toda la partición, pero esta es una manera rápida y fácil para proteger datos importantes.



Configuración.

Esta característica debe de ser configurada antes que la podamos utilizar ya que no se crea de manera automática con nuestra cuenta de usuario.

Básicamente son dos pasos de configuración:

1. Instalar la herramienta de cifrado (si no está instalada aún).

```
$ sudo aptitude install ecryptfs-utils
```

2. Configurar el directorio asignándole una contraseña o clave de cifrado a través

How-To

del siguiente comando:

```
$ ecryptfs-setup-private
```

En este paso preguntará la clave de login de la cuenta, para después asignar una clave de cifrado. Se puede dejar en blanco para que automáticamente genere esta clave.

Terminado el proceso, cerramos sesión y volvemos a loguearnos. En nuestra carpeta personal encontraremos el directorio Private al cual se puede acceder sin necesidad de introducir una contraseña.

Utilización.

En el directorio podemos añadir todos los datos que queramos mantener en privado, utilizándolo como un directorio normal.

Por ejemplo, si utilizamos Thunderbird, a fin de mantener privados nuestros correos podemos mover la carpeta a este directorio y crear un enlace simbólico para que el programa siga funcionando normalmente.

Primero movemos el directorio de Thunderbird al directorio Private:

```
$ mv ~/.mozilla-thunderbird ~/Private
```

Después hacemos un enlace simbólico que apunte a donde lo movimos:

```
$ ln -s ~/Private/.mozilla-thunderbird ~/.mozilla-thunderbird
```

Siempre tomando en cuenta que los datos que vayamos a añadir no los estemos utilizando con alguna aplicación. Tomando en cuenta el ejemplo anterior antes de mover el directorio debemos de cerrar el programa, en caso de tenerlo abierto.

Lo mismo podemos hacer con los directorios comunes de GNOME (Documentos, Música, Imágenes, etc) haciendo un enlace simbólico para mantenerlos siempre en primer plano y funcionando sin problemas en relación con aplicaciones que los usen.

Entrevista

Entrevista a Darien Alonso Camacho. Proyecto Cazabe.

A falta de flash cazabe, así define Darien Alonso Camacho a este maravilloso proyecto que se está realizando en nuestra isla, el cual pretende traer la animación y la interactividad vectorial al mundo del software libre y del código abierto.

Darien, quien gentilmente accedió ha esta entrevista nos contará un poco sobre el desarrollo de este proyecto que para muchos es desconocido, pero que promete un gran futuro dentro del mundo del software libre y el código abierto.

1. ¿ Quién es Darien Alonso Camacho ?

Soy Ingeniero en Ciencias Informáticas, graduado de la Universidad de Ciencias Informáticas (UCI) en el año 2009. Trabajo en el Joven Club #1 de Sagua la Grande, Villa Clara y mi labor tributa al Grupo de Desarrollo de Software Provincial (GDesPro VC) de la Empresa Joven Club de Computación y Electrónica. Me identifico con la filosofía del software libre, me gusta programar y compartir lo que hago.

2. ¿ Qué es realmente Cazabe ?

Cazabe es un proyecto joven que pretende traer la animación y la interactividad vectorial al mundo del software libre y del código abierto. Está implementado usando el lenguaje C++ y las bibliotecas Qt4. Su nombre procede de parafrasear el conocido refrán “a falta de pan cazabe” diciendo: “a falta de flash cazabe”. Tiene su sitio web oficial en



<http://www.vcl.jovenclub.cu/prov/cazabe/>

3. ¿ Cómo surge la idea de este proyecto ?

En el año 2009 fueron creados los Grupos de Desarrollo de Software en la entidad Joven Club de Computación y Electrónica (JCCE), ese mismo año terminé la universidad y me ubicaron dentro del grupo de Villa Clara. Se nos informó por parte de la dirección que el perfil de trabajo sería el desarrollo de software multimedia y video juegos. La plataforma flash es ampliamente usada en entornos Microsoft Windows para la creación de este tipo de programas, así que solo quedaba trabajar. En lo personal me interesaba hacer mi trabajo sobre GNU/Linux porque me gusta, por lo tanto intenté usar únicamente software libre para este propósito y al investigar qué opciones tenía llegué a la conclusión de que no existía una herramienta, distribuida bajo alguna licencia no privativa, lo suficientemente usable para cumplir en tiempo con mi trabajo. Decidí entonces emplear parte de mi tiempo en cubrir esta ausencia, sabía que era mucho trabajo para mi solo pero pensé “bueno, supongo que se me unan otros por el camino”. Tenía como antecedentes los proyectos Qflash, F4L y UIRA pero comencé mi implementación desde cero. En Noviembre de 2009 presenté el trabajo en mi municipio en la XII Edición del Evento de Informática para Jóvenes Infoclub 2010, de ahí pasó al evento provincial y luego

Entrevista

al nacional. Posteriormente Joven Club acogió Cazabe como proyecto oficial.

4.Para que los lectores entiendan. ¿ Se puede decir que Cazabe es como el flash de Adobe?

Sí se puede decir que Cazabe es como el flash de Adobe, pero solo para que los lectores entiendan. Las similitudes que puedan existir entre ambos persiguen disminuir la curva de aprendizaje del usuario procedente de Macromedia/Adobe Flash, nunca crear una copia fiel de ellos. El proyecto Cazabe desea separarse completamente del estereotipo que impuso Flash siguiendo las opiniones y sugerencias de la comunidad.

5.¿ Qué ventajas tendremos con el uso de este software?

Aunque debo decir que aún queda un largo camino “por codificar” ya veo muchas ventajas; Cazabe nos permitirá editar gráficos vectoriales y código Action Script en un mismo ambiente multiplataforma, totalmente libre y de código abierto, agilizando el trabajo de creación de software.

Podrá ser adaptado a las necesidades propias de quien lo deseé y evitará pagar considerables sumas de dinero por cada copia de su equivalente propietario. Sobre la base de lo alcanzado hasta el momento podremos innovar y brindar mejor experiencia a los desarrolladores de animaciones y aplicaciones interactivas. El software libre permite que seamos mejores “seres informáticos” porque podemos compartir sin violar las leyes y Cazabe es software libre.

6.¿Bajo qué Licencia se distribuye Cazabe?

Cazabe se distribuye bajo GPL, por el momento versión 2.0. La familia de licencias GPL me asegura que este se mantendrá libre por siempre y tengo particular interés en que esto suceda así.

7.¿ Cuántos miembros conforman el equipo

de desarrollo actualmente ?

Estoy solo en el equipo de desarrollo y me brinda colaboración voluntaria Viviana Rodríguez Martínez en cuanto a iconografía y propuesta visual. Quiero aprovechar este espacio para agradecer a todas las personas que han contribuido conmigo de las más disímiles formas, ya sea animándome para que continúe, opinando, facilitando trámites o poniendo su computadora personal de mejores prestaciones a mi disposición para determinados trabajos del proyecto.

8.¿ Están abiertos a recibir colaboración de otros desarrolladores?

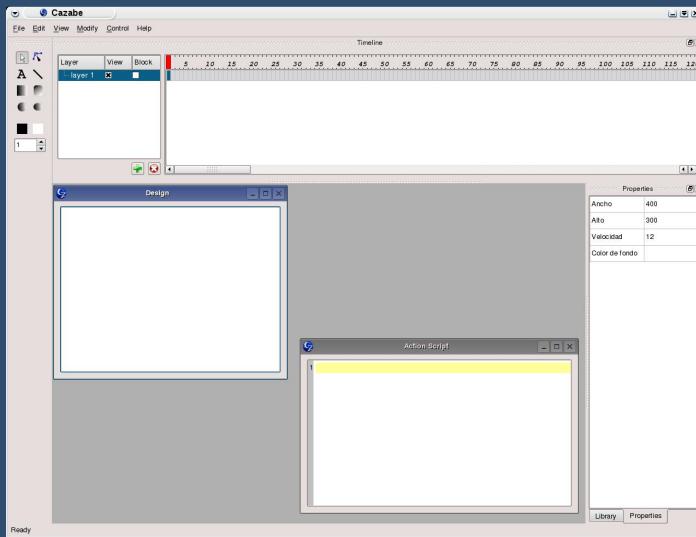
Por supuesto que sí, existen varias vías para colaborar. Una vez que se hayan publicado los fuentes, otros programadores podrán proponernos mejoras y nuevas funcionalidades, así como escribir documentación en la wiki del proyecto. En el sitio web intentamos mostrar toda la información en inglés y en español para facilitar la comprensión de nuestros visitantes, por lo tanto, siempre se requiere de personas con el interés de traducir los textos, cualquier interesado nos puede contactar.

9.¿ Para cuando verá la luz la primera versión de Cazabe?

La decisión de liberar un software por primera vez no es algo que se deba tomar a la ligera, todo el que alguna vez ha tenido que hacerlo estará de acuerdo conmigo. Espero poder poner Cazabe a disposición de todos muy pronto, aunque sea con calidad alpha. No voy a dar una fecha exacta pero trabajo para que suceda antes de un mes.

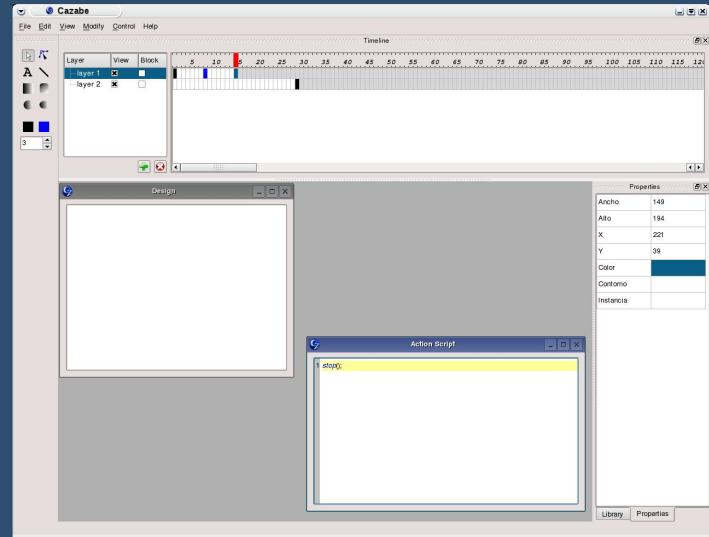
Entrevista

10. Podrías compartir algunas capturas de pantalla sobre Cazabe con nuestros lectores.

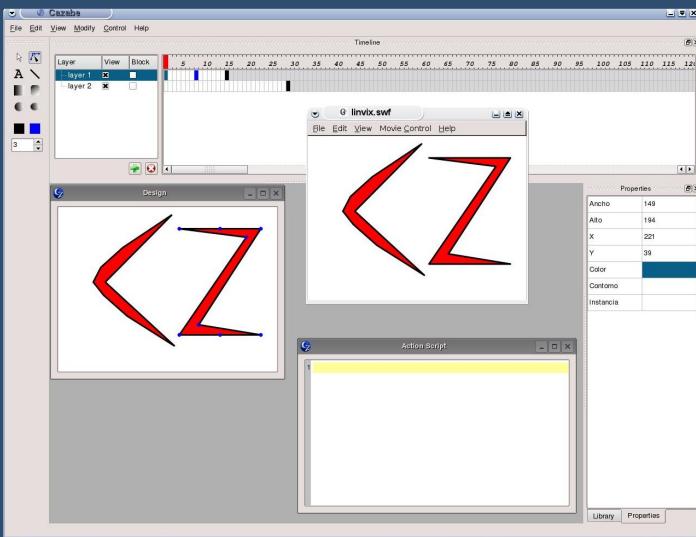


Interfaz principal.

forma fácil, solo con herramientas propietarias, será algo del pasado.



Inclusión en un fotograma de un script sencillo en lenguaje Action Script.



Inclusión y deformación de gráficos. El resultado es visualizado directamente con el reproductor libre de flash gnash.

11. Un mensaje a la comunidad

Este proyecto está dirigido a satisfacer las necesidades de la comunidad, sin ella carece de sentido, la opinión de cada usuario o programador cuenta, la comunidad es quien guía el desarrollo. Solo un poco más de espera y la creación de gráficos vectoriales interactivos de

Noticias

El cable de fibra óptica ya está en Santiago de Cuba

Ya llegó a las costas cubanas de Siboney, en la oriental ciudad de Santiago de Cuba, el cable submarino de fibra óptica procedente de Venezuela. Así lo informó a través de Telesur, Wilfredo Morales, presidente de la Empresa de Telecomunicaciones Gran Caribe.

Morales, indicó que este programa llevará la integración social a nuevos niveles sociales, económicos y culturales que permitirán el desarrollo comunicacional en Cuba, nación que desde hace 49 años es víctima de un criminal bloqueo por parte de los Estados Unidos.

“Con este cable que conecta a Cuba y Venezuela se impulsa la integración social, cultural y económica de los países de la región, pero va más allá, se trata de la complementariedad, solidaridad y cooperación entre los países”, destacó.

En el acto de oficialización de la llegada del cable submarino se encontraban el Comandante de la Revolución Ramiro Valdés Menéndez, Vicepresidente de los Consejos de Estado y de Ministros, Medardo Díaz, Ministro cubano de Informática y Comunicaciones y Manuel Fernández, Viceministro de Telecomunicaciones de la República Bolivariana de Venezuela.

El cable, en buena parte de su trayecto, reposará en el fondo marino, debido a que no existen corrientes de agua que lo muevan de su trazado original, solo en algunos puntos estratégicos se realizará un arado para enterrar el cable y mantenerlo fijo.

Fuente: Cubadebate

Presentarán Nova Linux, sistema operativo cubano



La versión 3.0 del sistema operativo cubano Nova Linux, desarrollado por jóvenes de la Universidad de Ciencias Informáticas (UCI), representa un paso de avance nacional hacia la sustitución de sistemas operativos extranjeros y costosos, fundamentalmente Windows, cuyo uso además, está vetado por las leyes del bloqueo.

El sistema operativo de producción cubano Nova Linux ha sido probado con éxito en equipos de procedencia extranjera, por lo que su capacidad de funcionamiento y compatibilidad está demostrada.

Según el documento oficial de lanzamiento, cuenta con las variantes de Nova Escritorio para minimizar el cambio brusco a que se enfrentan las personas familiarizadas con Windows. Nova Servidores para Servicios Telemáticos, Centros de Datos, Redes y Entornos de Desarrollo. Por último un Nova Ligero para computadoras con bajas prestaciones.

Se considera, además, un producto criollo “hecho por cubanos para cubanos, logrando inigualable adaptabilidad a las condiciones tecnológicas del país”

Boris Moreno, viceministro de Informática y Comunicaciones señaló recientemente que durante este año está previsto que las nuevas computadoras ensambladas en Cuba tengan instalados, tanto el sistema operativo Windows, como el Nova.

Fuente: Cubadebate

Nuevo logo para HTML 5



Pese a que HTML5 lleva tiempo entre nosotros, la adopción definitiva como estándar definitivo (no como desarrollo) entre programadores y desarrolladores web, todavía no ha llegado de forma oficial.

Con la intención de acelerar al máximo este proceso La World Wide Web Consortium ha presentado el logo oficial de HTML5, que nos permitirá identificar mejor a todos los desarrollos web que hagan uso del que para muchos es el futuro de Internet.

El logo representa no solo HTML5, sino CSS, SVG, WOFF y otras tecnologías que constituyen la plataforma web de código abierto para la creación de aplicaciones web y está disponible en:

www.w3.org

Noticias

Gnome 3 se presenta en sociedad con un sitio web

<http://gnome3.org/>



Ante las continuas noticias que salen de Unity, parece que en Gnome no han querido quedarse atrás y acaban de presentar el sitio web de Gnome 3. Pero no nos confundamos: se trata tan sólo de un sitio promocional en donde se habla de las bondades de este entorno de escritorio y poco más.

La página principal nos ofrece capturas de pantalla de como está a este momento Gnome 3 y, para que negarlo, se ve bastante bien. También tenemos un listado de sus principales características y un enlace para aquellos atrevidos que quieran probar la versión alpha, ya que todavía no hay algo que sea un poco más estable para los que no queremos correr tantos riesgos.

En la sección de preguntas frecuentes podemos leer que Gnome 3 estará listo para abril de este año, coincidiendo con el lanzamiento de Ubuntu 11.04. Y ojo, ojalá se cumpla y para entonces podamos disponer los usuarios de la distro de Canonical de ambos entornos para elegir el que mejor se adapta a nuestras necesidades.

Fuente: Muy Linux

Ubuntu abre paso a las librerías Qt

En un comunicado de Mark Shuttleworth nos sorprende encontrar la noticia de que Ubuntu abre paso a las librerías Qt y está buscando la forma de hacerles un hueco en el espacio que ocupa la instalación del sistema en el Live-Cd para incluirlas por defecto en la siguiente lanzamiento a Ubuntu Natty, es decir, en Ubuntu 11.10.

La justificación para esta inclusión de las librerías utilizadas por KDE en Ubuntu, que recordemos lleva escritorio Gnome y por lo tanto librerías GTK, se basan en la filosofía de búsqueda de selección de aplicaciones para Ubuntu donde se tiene en cuenta que sean buenas para los usuarios, con una buena usabilidad y que faciliten el uso de Linux al usuario, por lo tanto sin distinguir entre aplicaciones para KDE y para Gnome.

Esta decisión no significa que en la próxima distribución de Ubuntu veremos Amarok, como reproductor de música por defecto. Más bien me parece que están tendiendo puentes para poder incorporar en el futuro alguna aplicación de KDE.

Precisamente han tenido cuidado de no molestar a la comunidad de Gnome y a la vez dejar la pelota en el tejado de la comunidad de KDE para que una vez realizado el trabajo para conectar las librerías QT con GTK sean ellos los que contribuyan también en la integración.

DuckDuckGo, buscador que respeta la privacidad del usuario

<http://duckduckgo.com/>

DuckDuckGo es un motor de búsqueda cuyo leitmotiv es el respeto a la privacidad del usuario al no recoger información sobre éste que lo identifique. Detrás del proyecto está Gabriel Weinberg, un licenciado en física y master en Tecnología en el MIT que saltó a la fama por un desarrollo sobre bases de datos de nombres que vendió posteriormente por una cifra astronómica a United Online.

El proyecto es de código abierto y está escrito en Perl, alojado en un servidor Nginx ejecutándose sobre FreeBSD, que también son tecnologías de código abierto.

DuckDuckGo no recopila o comparte información personal. Esa es nuestra política de privacidad en pocas palabras. El motor de búsqueda DuckDuckGo es rápido y ofrece resultados similares en una búsqueda que cualquier otro. Como inconvenientes, idioma inglés como única alternativa y obviamente está muy lejos de proporcionar los servicios a los que estamos acostumbrados.



[About](#) | [Settings](#) | [Goodies](#) | [Add to Firefox](#)

© 2011 - We don't track you! - Also at duckgo.com

Noticias

LibreOffice 3.3 lanzado oficialmente

The Document Foundation finalmente ha lanzado la versión inaugural estable de LibreOffice, la bifurcación comunitaria de OpenOffice creada para garantizar la continuidad de este paquete ofimático, a raíz de las controvertidas acciones de Oracle con algunos proyectos anteriormente pertenecientes a Sun.



LibreOffice

The Document Foundation

Algunas de las mejoras que incorpora LibreOffice 3.3 son la importación y edición de gráficos SVG, una herramienta de navegación mejorada para Writer, una versión mejorada de Calc (con soporte para un millón de filas), importación de PDF, mejoras en la protección de documentos de Write y Calc, mejoras en la interfaz de impresión, y otras.

LibreOffice 3.3 puede descargarse para todas las plataformas principales y está respaldado por más de 100 desarrolladores desde que el proyecto se conformó, el año pasado. En la versión para Windows se ha integrado todos los idiomas a fin de evitar múltiples descargas.

Fuentes: Phoronix, The Document Foundation

LibreOffice ya forma parte de Ubuntu 11.04

Mark Shuttleworth había dicho que sí, que esto pasaría; luego se confirmó pero no se daban fechas. Y ha sido con la actualización de hoy que los usuarios que están probando Natty se han encontrado con ello: Libre Office ha reemplazado a OpenOffice como la suite ofimática por defecto de Ubuntu.

LibreOffice todavía anda en sus versiones Release Candidate, pero parece que desde Canonical han juzgado que la suite ya está lo suficientemente madura como para incluirla en Ubuntu 11.04.

Entre las distros más populares, Ubuntu no va a ser la única que va realizar este cambio, sino que también Fedora anunció que incluirá LibreOffice en su versión 15.

Fuente: Muy Linux

El kernel 2.6.38, con mejoras de rendimiento de hasta el 50%

Se acaba de lanzar la primera versión Release Candidate del núcleo Linux 2.6.38, y entre las novedades está una mejora de rendimiento sustancial gracias a la eliminación de uno de los grandes "cerrojos" que limitaban ciertos apartados.

En el kernel 2.6.37 se eliminó por fin el llamado Big Kernel Lock (BKL) pero aparentemente había al menos otro cerrojo global que necesitaba salir del núcleo lo antes posible. Se trata del llamado RCU (Read/Copy/Update) que se usa en la búsqueda de los nombres de ruta.

El propio Linus Torvalds comentaba la situación:

"La eliminación del RCU, es material realmente bueno, y se encargará de hacernos olvidar el último de los principales cerrojos globales que realmente tienden a dañar algunas cargas en el núcleo".

Torvalds añadió además que el nuevo mecanismo de búsquedas de rutas mejorará el rendimiento del kernel entre un 30% y un 50% según la situación.

Además hay una nueva característica de autoagrupado en el planificador que Torvalds también incluyó entre sus novedades favoritas en la publicación del kernel 2.6.38 RC1.

Esto permite ofrecer sesiones tty con sus grupos separados, lo que según Linus proporciona "un comportamiento interactivo más apropiado". De hecho, afirmaba el creador del kernel, "si aún haces trabajo de verdad en un terminal, vas a apreciar mucho esta mejora".

Noticias

¡Firefox 4, 5, 6 y 7 en 2011!

A pesar de lo que ha sucedido hasta ahora, el desarrollo de las próximas versiones del navegador de Mozilla va a cambiar de estrategia de forma radical, y en 2011 no solo veremos Firefox 4, sino también Firefox 5, Firefox 6 y Firefox 7.



La documentación de Mozilla y de Firefox en formato wiki así lo confirma, algo sorprendente teniendo en cuenta que desde que apareció Firefox 3.0 han pasado más de 2 años y medio. La idea es la presentar versiones menos ambiciosas y con más rapidez.

La documentación de Mozilla ha dejado entrever ese cambio de actitud que hará que en contra de lo que hemos visto hasta ahora, se lancen versiones finales del navegador con un gran salto en el número de versión en lo que resta de 2011.

De hecho en pocas semanas veremos Firefox 4 por fin, pero Mozilla ya está preparando el lanzamiento de Firefox 5, Firefox 6 y Firefox 7, que llegarán también durante 2011 y lo harán, claro está, con un conjunto de cambios diferenciales mucho menor que el que afecta al salto que se ha dado de Firefox 3 a Firefox 4.

Como explican en esa documentación, en Firefox 5 dispondremos de un gestor de cuentas y una serie de novedades en el terreno de la “navegación social”, algo que Mozilla comenzó a investigar con la extensión F1 de Mozilla Labs. Además, parece que habrá una versión específica de Firefox 5 para Windows 7 64 bits.

En Firefox 6 los responsables de Mozilla se centrarán en la puesta en marcha definitiva del framework para sus aplicaciones web (el proyecto Chromeless, antes conocido como Prism), además de optimizaciones al motor JavaScript y soporte para Mac OS X 10.7.

En cuanto a Firefox 7, la lista de novedades estó menos clara, pero podría aplicarse la tecnología ya bautizada como Electrolysis, que aplicará en Firefox lo que Google Chrome viene haciendo desde hace tiempo: separar pestañas de navegación como procesos distintos.

Llegó Debian 6.0 Squeeze cargadita de regalos y nueva web

La versión final estable de Debian 6.0 acaba de llegar al mercado tras dos largos años de desarrollo. Debian es la madre de muchas distribuciones y si bien no está tan actualizada como sus hijas sí que es claro ejemplo de estabilidad.

La versión 6.0 es conocida como Squeeze y ofrece uno de los mayores soportes multiplataforma existentes, pudiendo trabajar en sistemas x86, ARM, 32 y 64 bits, MIPS y una larga lista de plataformas hardware.

Algunas de las novedades que nos podemos encontrar:

- * El entorno de escritorio KDE 4.4.5
- * El entorno de escritorio GNOME 2.30
- * El entorno de escritorio Xfce 4.6
- * El entorno de escritorio LXDE 0.5.0
- * X.Org 7.5
- * OpenOffice.org 3.2.1
- * GIMP 2.6.11
- * PostgreSQL 8.4.6
- * MySQL 5.1.49
- * Linux 2.6.32
- * Apache 2.2.16
- * Y más de 29,000 paquetes de programas listos para usarse construidos a partir de 15.000 paquetes fuente.

Además de la renovada web Debian.org desde donde podrán conseguir las imágenes de esta nueva versión



El Personaje

Richard Matthew Stallman

Richard Matthew Stallman nacido en Manhattan, Nueva York el 16 de marzo de 1953 es un programador estadounidense y fundador del movimiento por el software libre en el mundo.

Entre sus logros destacados como programador se incluye la realización del editor de texto GNU Emacs, el compilador GCC, y el depurador GDB, bajo la rúbrica del Proyecto GNU. Sin embargo, es principalmente conocido por el establecimiento de un marco de referencia moral, político y legal para el movimiento del software libre, como una alternativa al desarrollo y distribución del software no libre o privativo. Es también inventor del concepto de copyleft (aunque no del término), un método para licenciar software de tal forma que su uso y modificación permanezcan siempre libres y queden en la comunidad.

En 1971, siendo estudiante de primer año de Física en la Universidad Harvard, Stallman se convirtió en un hacker del laboratorio de Inteligencia artificial (IA) del Instituto Tecnológico de Massachusetts (MIT). En los años 1980, la cultura hacker que constituía la vida de Stallman empezó a disolverse bajo la presión de la comercialización en la industria del software. En particular, otros hackers del laboratorio de IA fundaron la compañía Symbolics, la cual intentaba activamente reemplazar el software libre del Laboratorio con su propio software privativo.

Durante dos años, desde 1983 a 1985, Stallman por sí solo duplicó los esfuerzos de los programadores de Symbolics para impedir que adquirieran un monopolio sobre los ordenadores del laboratorio. Por ese entonces, sin embargo, él era el último de su generación de hackers en el laboratorio.



Se le pidió que firmara un acuerdo de no divulgación (non-disclosure agreement) y llevara a cabo otras acciones que él consideró traiciones a sus principios. El 27 de septiembre de 1983 Stallman anunció en varios grupos de noticias de Usenet el inicio del proyecto GNU que perseguía crear un sistema operativo completamente libre.

Al anuncio inicial del proyecto GNU siguió, en 1985, la publicación del Manifiesto GNU, en el cual Stallman declaraba sus intenciones y motivaciones para crear una alternativa libre al sistema operativo Unix, al que denominó GNU (GNU No es Unix), pronunciado de forma parecida a ñu, en inglés. Poco tiempo después fundó la organización sin ánimo de lucro Free Software Foundation para coordinar el esfuerzo. Inventó el concepto de copyleft, que fue utilizado en la Licencia Pública General GNU en 1989. La mayor parte del sistema GNU, excepto el núcleo, se completó aproximadamente al mismo tiempo.

En 1991, Linux Torvalds liberó el núcleo Li-

El Personaje

nux bajo los términos de la GPL, completando un sistema GNU completo y operativo, el sistema operativo GNU/Linux.



Las motivaciones políticas y morales de Richard Stallman lo han convertido en una figura controvertida. Muchos programadores influyentes que se encuentran de acuerdo con el concepto de compartir el código, difieren con las posturas morales, filosofía personal o el lenguaje que utiliza Stallman para describir sus posiciones. Un resultado de estas disputas condujo al establecimiento de una alternativa al movimiento del software libre, el movimiento de código abierto.

"Que las empresas tengan especial influencia en la política significa que la democracia está enferma. El propósito de la democracia es asegurarse de que los ricos no tengan una influencia proporcional a su riqueza. Y si tienen más influencia que tú o que yo, eso significa que la democracia está fallando. Las leyes que obtienen de esta forma no tienen autoridad moral, sino la capacidad de hacer daño."

En 1999 promovió la creación de una enciclopedia libre, la GNUPedia, considerada como un antecedente directo de la Wikipedia. En la actualidad, Stallman se encarga de difundir la ideología GNU en todo el mundo mediante charlas y conferencias. Esto hace que algunos le consideren como un profeta o gurú.

Stallman posee un buen nivel de castellano, y ha dado numerosas conferencias en español en Argentina, Cuba, España, Perú, México, Venezuela, Uruguay, Colombia, y otros países latinoamericanos. Durante sus charlas aparece con una segunda personalidad que corresponde a San Ignacio en la que suele bendecir las computadoras de las personas.

Su aparentemente descuidada imagen y sus manías extravagantes (no utiliza teléfono móvil, siempre va acompañado de su computado-

ra portátil, no duerme en hoteles, etc.) le han hecho ser blanco de numerosos chistes y bromas, llegando a aparecer en tiras cómicas.

Stallman ha recibido numerosos premios y reconocimientos por su trabajo, entre ellos:

- Una membresía en la MacArthur Foundation en 1990.
- El Grace Hopper Award de la Association for Computing Machinery en 1991 por su trabajo en el editor Emacs original.
- Un Doctorado Honoris Causa del Royal Institute of Technology de Suecia en 1996.
- El Pioneer award de la Electronic Frontier Foundation en 1997.
- El Yuri Rubinsky memorial award en 1999.
- El Takeda award en 2001.
- En 2004 recibió un Doctorado Honoris Causa otorgado por la Universidad de Salta (Argentina).
- Primer galardonado con el Premio Internacional Extremadura de Conocimiento Libre que concedió la Junta de Extremadura el 7 de febrero de 2007 en Badajoz en el marco de la III Conferencia Internacional de Software Libre.
- Doctor Honoris Causa de la Universidad de Los Ángeles de Chimbote, en Perú en 2007
- Doctor Honoris Causa de la Universidad de Pavia, en 2007
- Doctor Honoris Causa de la Universidad Nacional de Trujillo, en Perú en 2008
- Doctorado Honorario en Ciencias en la Universidad de Lakehead en Canadá en 2009

Fuentes consultadas
<http://es.wikipedia.org/>



Linvix

Participa en la revista, envíanos
tus artículos, y opiniones ...

linvix@gmail.com

El conocimiento
pertenece al
Mundo.

