
Viejos ordenadores que hacen grandes cosas

#Servidor Local – Debian GNU/Linux



Manual original, creado por David Sanchez Cantero.
En la red (Forat)

ForatDot.Info

Este **servidor** no es capaz tan solo de servir *contenido web* si no que también *comparte* sus *discos duros e impresoras* conectadas a el. *Servidor DNS y gestor de descargas* para navegar mas deprisa y que alguien se ocupe de descargar lo que necesitemos. Por si esto fuera poco le conectamos varias tarjetas de red y montamos un *Switch de red* para así ahorrarnos un cacharro menos conectado a la red. El router se conecta a una tarjeta de red y los demás ordenadores que tengamos se conectan a las tarjetas de red libres.



A continuación podemos ver el temario ...

Introducción

- Vol 1 – [Como instalar Debian](#)
- Vol 2 – [Como montar un switch de red](#)
- Vol 3 – [Como montar un servidor DNS](#)
- Vol 4 – [Como montar un servidor web](#)
- Vol 5 – [Como montar un servidor NFS](#)
- Vol 6 – [Gestor de descargas vía web](#)
- Vol 7 – [Como montar un servidor de impresión](#)
- Vol 8 – [Web de control y phpSysInfo](#)

(Introducción)



Este verano me prometí no terminarlo sin montarme algo guapo con todo el **Hardware** decente que tengo por aquí almacenado. Desde hace muchos años tengo uno o dos servidores funcionando las 24 horas en mi cuarto.

En ellos suelo tener de manera estable alguna web y algunas de las imágenes que podéis ver en esta web. Siempre me digo de montar algo practico para el exterior y el interior de mi cuarto pero siempre acabo por tener un servidor web con lo básico (*Apache, MySQL, SSH*) y poco mas para que ofrezca servicio web al exterior.

Como este año me he prometido montarme algo guapo voy a compartirlo con todos vosotros y si tenéis algún ordenador por hay que queráis usar para algo practico pues tengáis una guía que os acompañe en el camino ...

¿ Que vamos a montar ?



Lo que vamos a montar es un pupurrí de cosas que me han parecido necesarias para mi entorno de trabajo. Tengo varios servidores como este con mas o menos servicios instalados pero todos cumplen su función.

En casa tengo este servidor a completo y hace de **servidor web**, **servidor de descargas**, **servidor de archivos NFS**, **servidor de impresión**, **servidor DNS Local** y **switch de red**.

Este servidor puede incorporar todos los servicios que vamos a ver a continuación en conjunto como por separado según vuestras necesidades ...

¿ Sobre que Hardware ?

Procesador : Pentium 4 a 3Ghz

Memoria ram : 2Gb

Red : 3 tarjetas de red + 1 integrada en la placa base

Lector : DVD

Discos : 1 x 80Gb + 1 x 160Gb

Basándonos en mi ordenador sabemos que con este **Hardware** es suficiente para que todo funcione correctamente y fluido. Con **Hardware** mas actual todo ira mejor, mas rápido y si tus piezas son inferiores en principio no hay problema pero no puedo asegurar su correcto funcionamiento mas que nada porque no lo he probado.

Este ordenador será un servidor y como tal no tendrá monitor conectado ni entorno gráfico ni ratón. Tan solo estará conectado a la corriente y a tu red así que todo lo que vamos a instalarle lo haremos a base de comandos.

Como no voy a tener monitor necesito que todo pueda manejarlo remotamente desde el navegador o la terminal. Este ofrecerá distintos servicios hacia nuestra red de área local como hacia Internet.

Sistema Operativo



Para montar este **servidor** vamos a instalar como sistema operativo **Debian GNU/Linux 6**, sin entorno gráfico, con lo justo y necesario para tener una instalación mínima. Siempre que montamos un servidor es interesante instalarle lo mínimo y así nosotros podemos instalarle uno a uno los servicios que queramos. Si elegimos una versión mínima como es la **NetInst** que además de necesitar menos tiempo de descarga nos instalará lo mínimo sobrecargando menos la maquina.

Servidor DNS Local

Como todos sabéis la tarea de un **servidor DNS** es traducir nombres de dominio a direcciones IP. Podemos usar montar un **servidor DNS** para alojar varios dominios en nuestro servidor o podemos usarlo de forma local para que cumpla la misma función que nos ofrece nuestro **ISP** (*Proveedor de servicios de Internet*).

¿ Cuantas veces nos hemos quedado esperando a que se resuelva la **DNS** al navegar porque nuestro **DNS** del **ISP** esta algo OUT ? Yo varias pero esto se ha terminado.

Cuando introducimos un nombre de dominio en el navegador y pulsamos la tecla (*Enter*) el navegador mira si tiene la web que pedimos en la cache, de no ser así envía la petición al **servidor DNS** del **ISP**, este si la tiene en la cache la sirve y si no pues la tarea se la encargara al **servidor DNS**.

No se cual es vuestro ancho de banda pero yo tan solo tengo una ADSL de 3 Mb en casa y mi red local funciona a 10/100Mbps con lo que ganaremos mucho mas en tiempo de respuesta si tenemos este servicio de manera local que no remota. Para montar el **DNS Local** usaremos el **servidor BIND** yo lo tengo hace tiempo funcionando en casa y se nota bastante en la velocidad a la hora de navegar.

Servidor Web

Ya que lo vamos a tener conectado las 24 horas seguramente tenemos algún blog o web que ofrecer al publico. Este **servidor web** es capaz de aguantar bastante trafico si esta correctamente configurado, todo dependerá de la velocidad de la red tanto local como exterior. En el apartado del servidor web instalaremos **Apache Web Server**, **MySQL**, **PHP** y para estadísticas del servidor el mítico **phpSysInfo** todo en sus ultimas versiones.

Servidor de archivos NFS

En mi casa no tenemos ningún ordenador que no use **GNU/Linux** así que un **servidor de archivos NFS** será mas que suficiente para que todos los ordenadores que tenga conectados tengan acceso a sus discos duros con un simple click que tendremos en el escritorio y listo. ¿ Tienes varios discos duros ? Conéctaselos a tu nuevo servidor y úsalos de almacén que compartir con todos tus ordenadores.

Switch de red

Si tenemos varios ordenadores conectados a un **Switch** y este al **Router** podemos evitarnos el **Switch** instalándole varias tarjetas de red, tantas como ordenadores tengamos o como ranuras PCI tenga nuestra placa base.

Con esto nos evitamos tener conectado otro aparato mas consumiendo electricidad y sobre todo disminuimos en cables ya que en vez de ir el **Router** conectado al **Switch** y del **Switch** al resto de ordenadores conectaremos directamente el **Router** a nuestro nuevo servidor y el resto de ordenadores se conectarán a el. Es importante que vuestras tarjetas de red funcionen a un mínimo de 10/100 Mb como mínimo y si son mas rápidas mejor que mejor.

Servidor de descargas

Todos hemos querido alguna vez que nuestro servidor se encargará de descargar lo que necesitemos y podamos decírselo desde un entorno web. **PyLoad** es un excelente **gestor de descargas** el cual posee un **interface web** desde el que podremos manejarlo prácticamente al completo. Llevo unas semanas usándolo y funciona muy bien sobre todo con las descargas de **Megaupload**, al igual que **JDownloader** espera automáticamente los 40 segundos, se salta el CAPTCHA y comienza la descarga. También lo he usado para descargas directas y funciona al 100%. Tiene la opción de que incluyas tu usuario premium y es compatible con otros servicios de descarga que hay por la red.

Como también nuestro servidor va a compartir archivos por NFS con toda nuestra red de ordenadores es súper cómodo incluir la ruta de descarga de **PyLoad** a la unidad que compartamos.

Una vez hecho esto tan solo tendremos que arrancar nuestro portátil, conectarnos a la web de **PyLoad**, agregar lo que queramos descargar y listo. Cerramos el navegador y nuestro servidor se ocupará de todo lo demás. Cuando queramos ver lo que nos descargo con acceder a nuestra unidad compartida por NFS, hacemos click y a disfrutar !

Servidor de impresión

Si tenemos una o varias impresoras y varios ordenadores conectados a la red estaría bien que en vez de ir conectando de un ordenador o otro el cable de la impresora, la conectáramos a nuestro servidor y que este se ocupara de compartir la impresora con lo demás ordenadores que tengamos. Esto es útil en cualquier casa pero imaginad si tenemos una clase con 50 ordenadores, si compartimos la impresora todos pueden imprimir sin tener que conectar nada, tan solo tendrán que imprimir lo que quieran, se comunicarán por la red con el servidor y este imprimirá gracias al **servidor de impresión CUPS** que instalaremos y configuraremos, bonito, bonito ...

Web de control

Para finalizar el **servidor local** vamos a montar una pequeña web para poder acceder a los servicios del **servidor** de una manera mas centralizada. En esta pequeña web de control tendremos las opciones necesarias para acceder desde cualquier navegador al *servidor de descargas, la terminal, información del sistema, administración de impresoras, Bases de datos, nuestras webs, etc ...*

La finalidad del proyecto

La finalidad del proyecto es simplemente aprender como funciona el sistema operativo **Debian GNU/Linux** y las aplicaciones que instalemos. Claro está que va a ser guapo tener nuestra maquina equipada con cosas como un **servidor de descargas vía web** o el **servidor de impresoras** que compartirá nuestra **impresora** con todos los ordenadores además de nuestros discos duros por **NFS** que tengamos pero lo realmente importante es el aprender a movernos por el sistema y saber que estamos haciendo.

Vol 1 - Como instalar Debian GNU/Linux



Comenzamos con el proyecto ([Viejos ordenadores que hacen grandes cosas](#)) ahora que tenemos nuestros ordenadores preparados para jugar. No pensaba incluir este artículo sobre la instalación de **Debian** pero ya que cambiado de la versión 5 o a la 6 he pensado en crear el tutorial por si hay algunos cambios y por si alguien lo necesita. Como siempre no perderemos detalle con imágenes y explicaciones sobre lo que estamos haciendo ...



Cabe decir que para este proyecto usaremos un disco duro entero para el sistema operativo y así complicaremos aun menos la historia. Tan solo vamos a crear una partición (/) y otra (SWAP) que son las básicas que necesitamos para poder montar este (**Servidor Local**). De todos modos cuando compartamos unidades por NFS con nuestra red de ordenadores podremos montar mas discos y usarlos como esclavos.

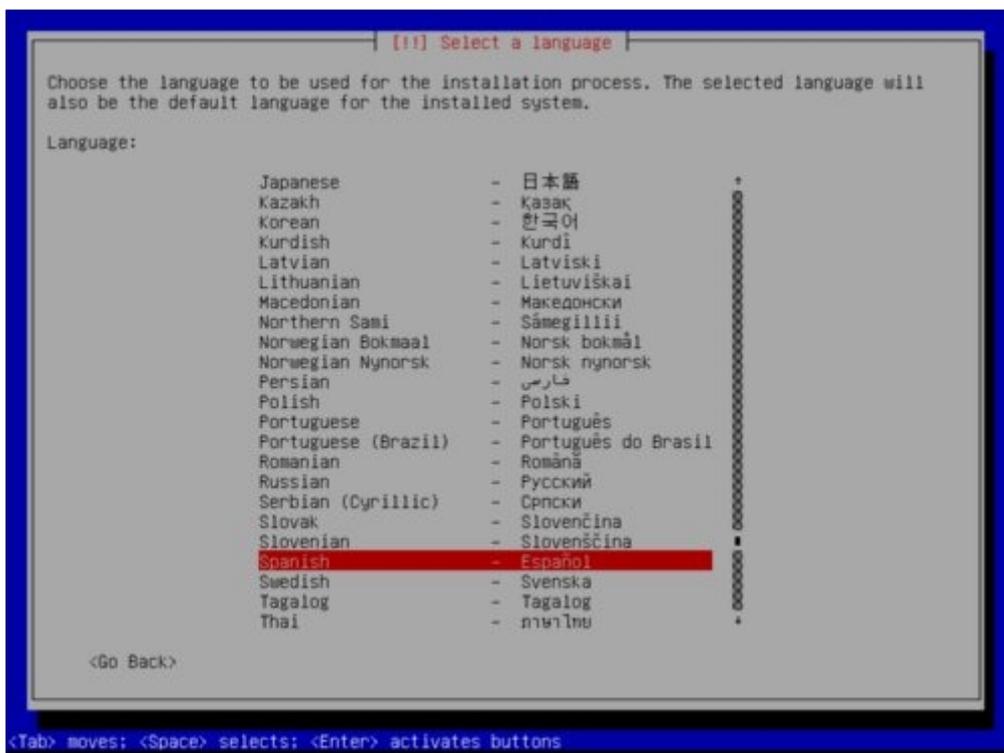
Una vez sabido esto vamos a descargarnos el sistema operativo **Debian 6** en su versión **NetInst**. Esta es una versión que tan solo ocupa 191 Mb y necesita conexión a internet para su correcta instalación.

Aquí os dejo los enlaces directos para su descarga en las dos vertientes mas usadas de procesadores que son [32](#) y [64](#) Bits ...

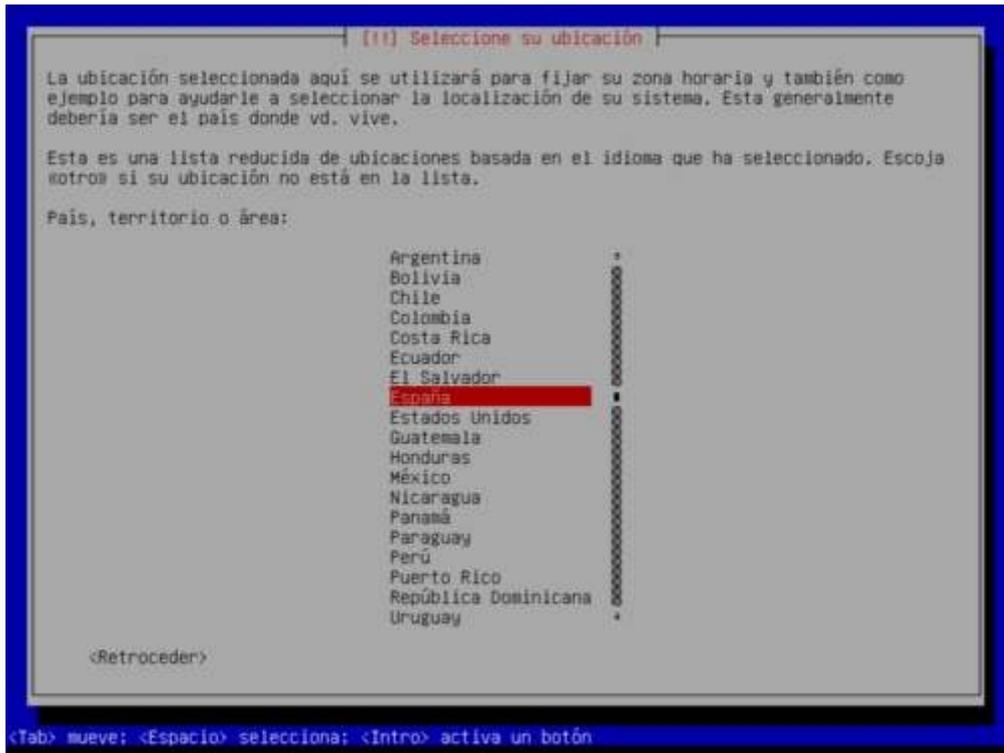
Una vez descargada nuestra versión la grabamos en un CD como disco de imagen e insertamos el CD, arrancamos nuestro nuevo (**Servidor Local**) y veremos lo siguiente ...



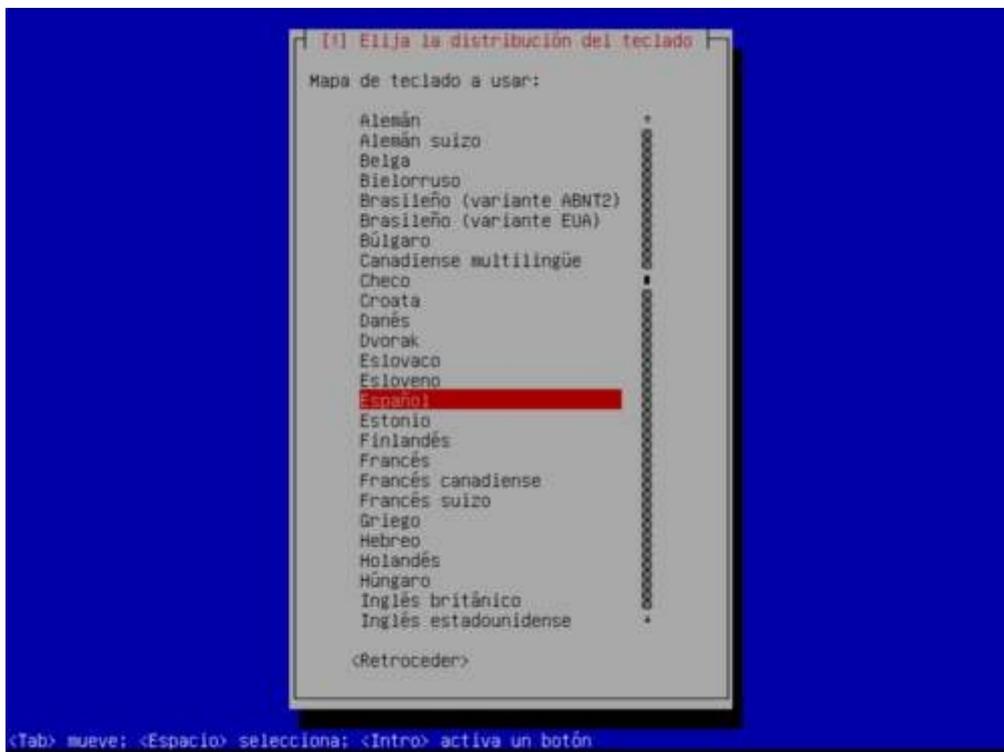
Seleccionamos la opción (*Install*) y continuamos con la tecla (*Enter*) ...



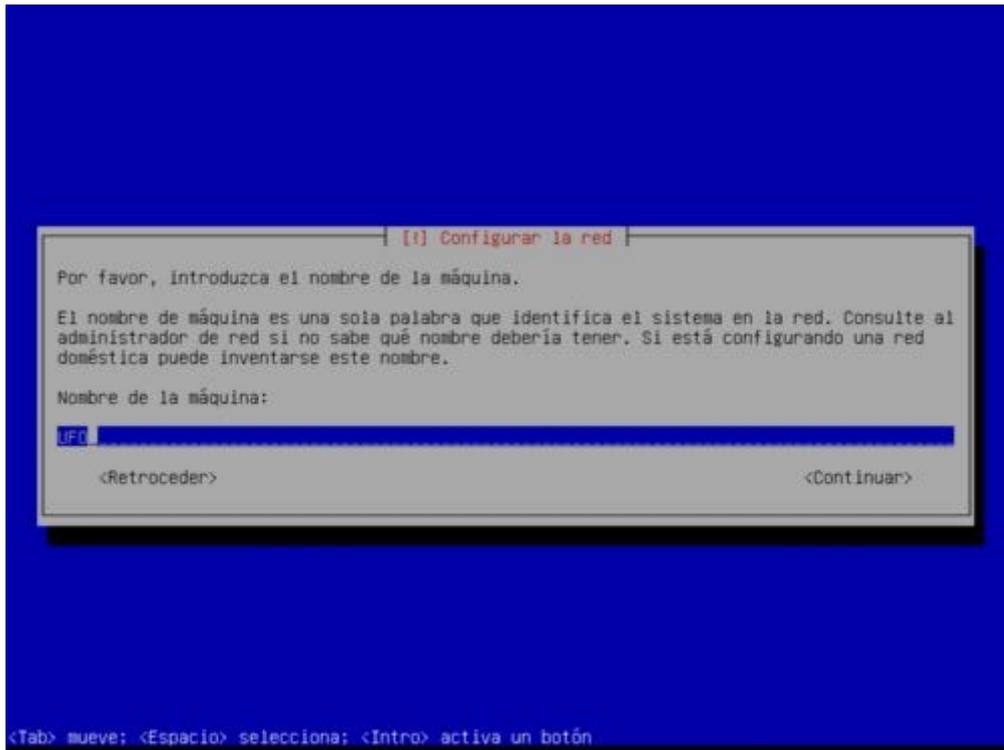
Seleccionamos nuestro *idioma* y pulsamos la tecla (*Enter*) ...



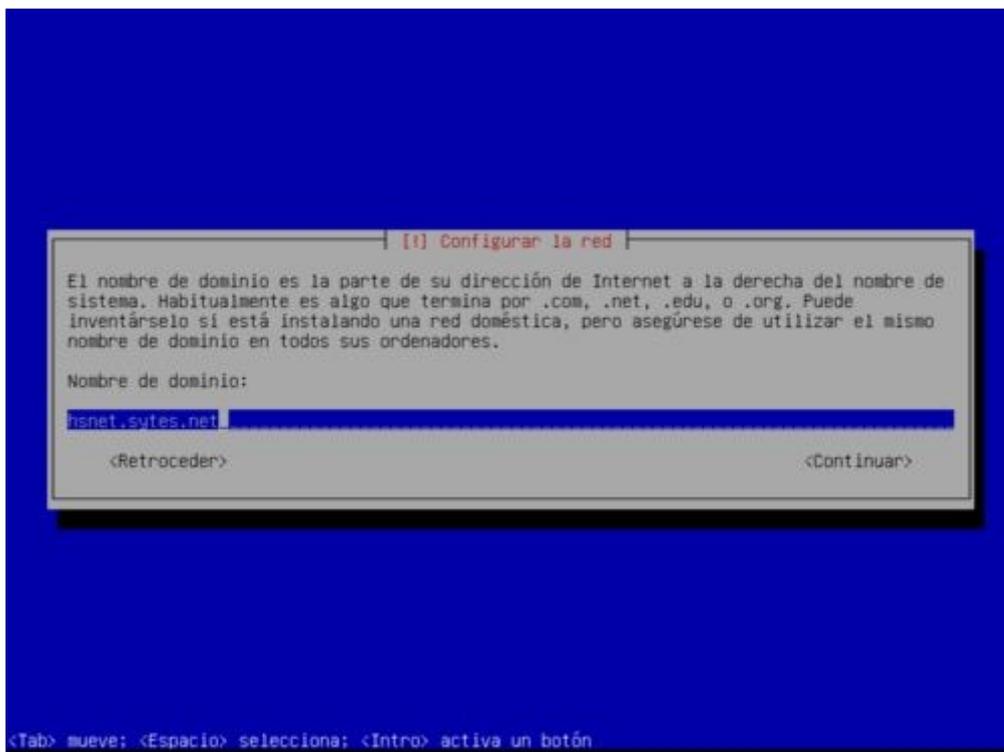
Seleccionamos nuestro *País* y pulsamos la tecla (*Enter*) ...



Seleccionamos el *Idioma* de nuestro teclado y pulsamos la tecla (*Enter*) ...



Introducimos un nombre para nuestro servidor, en mi caso le he puesto (*UFO*), tu ponle el que quieras. Una vez hecho esto continuamos pulsando la tecla (*Enter*) ...

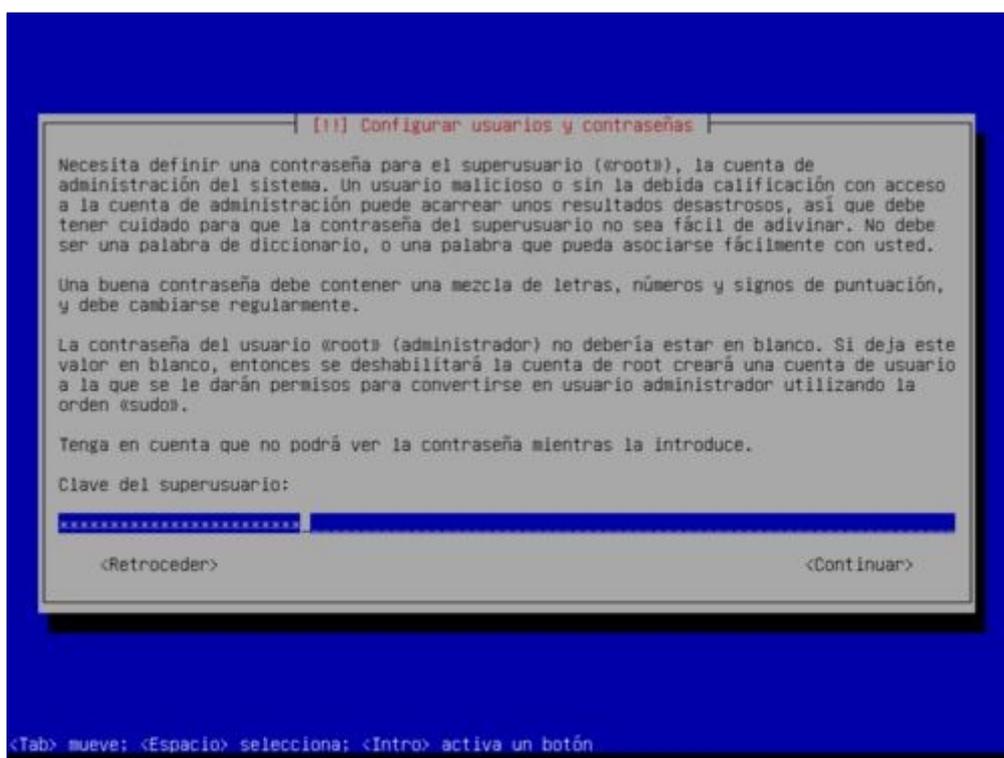


Introducimos un nombre de dominio. En mi caso he puesto (hsnet.sytes.net) porque lo tengo redireccionado hacia mi **Router**. Los que estáis por aquí desde hace años seguramente conoceréis el

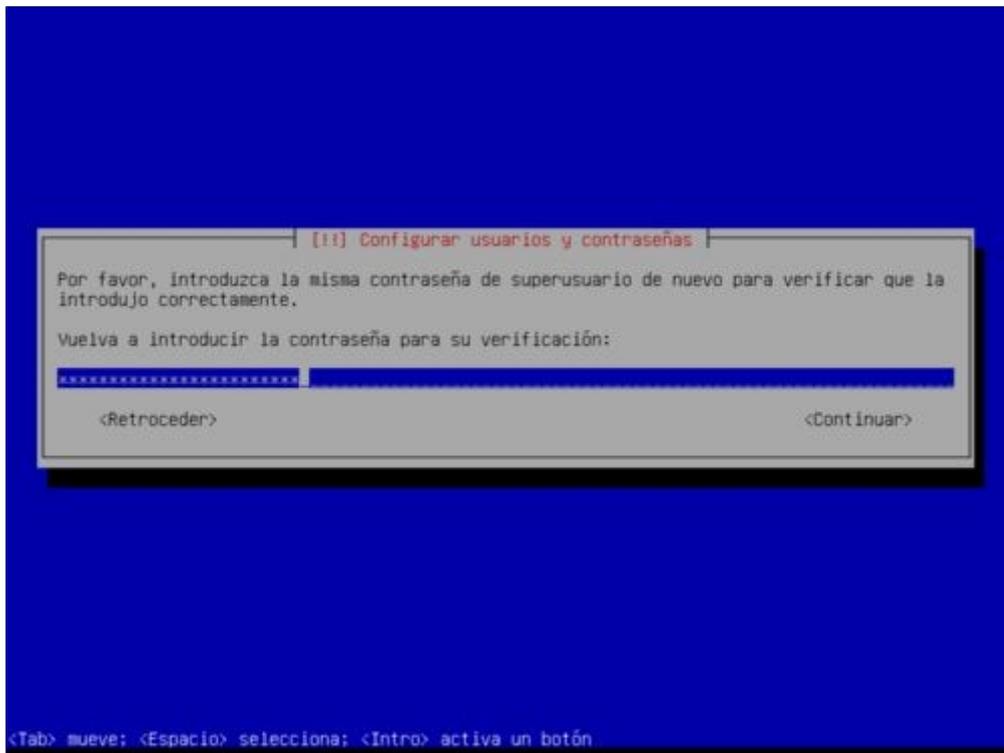
servicio gratuito de NO-IP, para los que llegáis recientemente y no lo conocéis os aconsejo que echéis un vistazo al tutorial ([Encontrando nuestro servidor desde Internet con No-IP](#))

Si no tenéis ningún servicio de redirección IP pero tenéis un dominio comercial relacionado con vuestro servidor podéis ponerlo aquí. Si no tienes nada pon el dominio que quieras, puedes cambiarlo mas adelante en sus archivos de configuración.

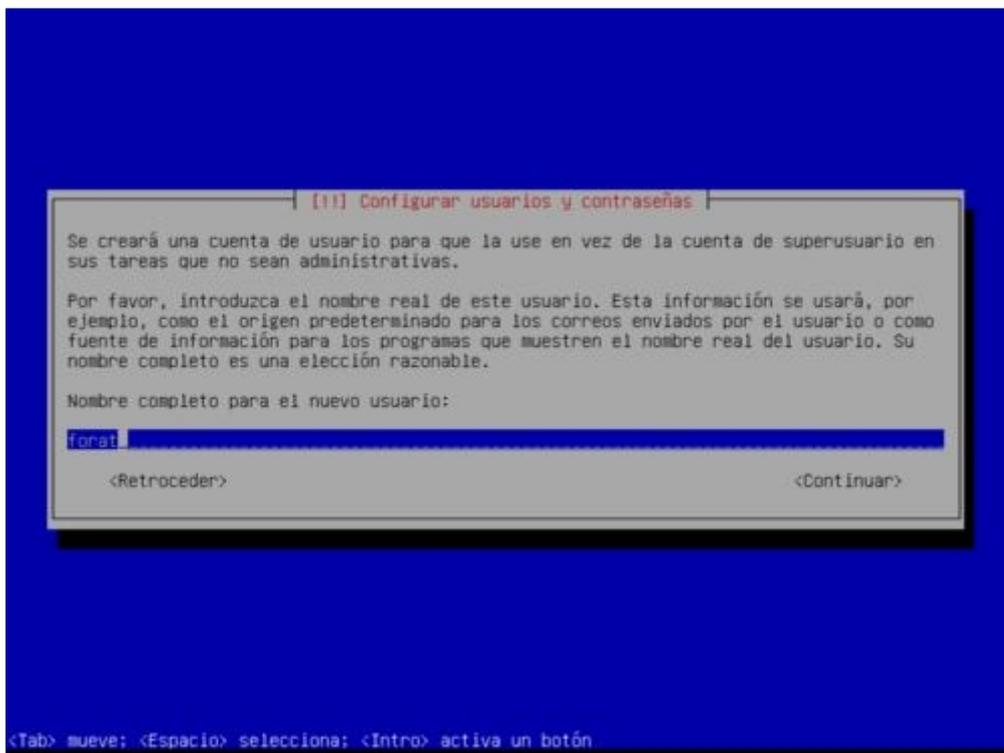
Una vez introducido el nombre de *dominio* pulsamos la tecla (*Enter*) ...



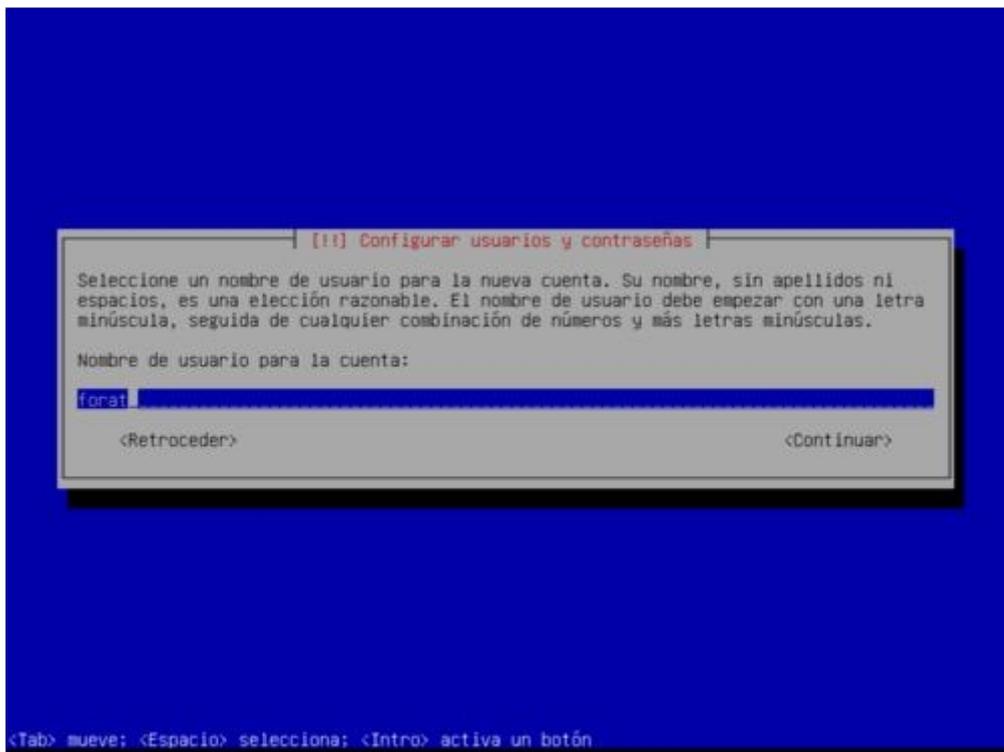
Introducimos una contraseña para el súper usuario (*root*) y pulsamos la tecla (*Enter*) ...



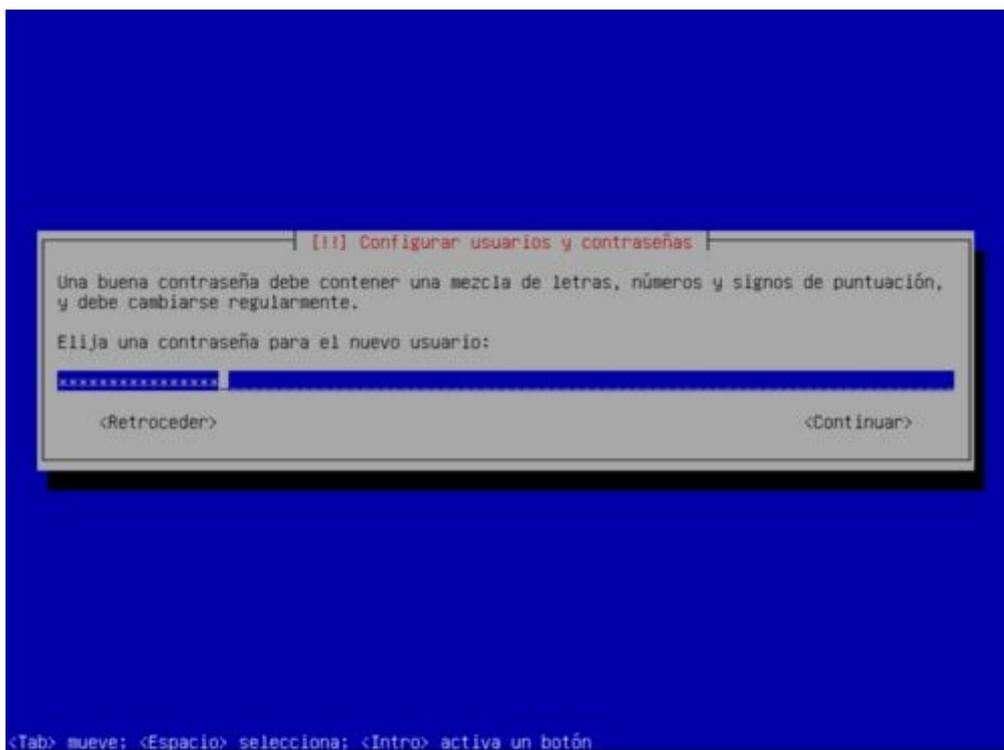
Introducimos de nuevo la misma contraseña para *verificar* que la hemos escrito bien y pulsamos la tecla (*Enter*) ...



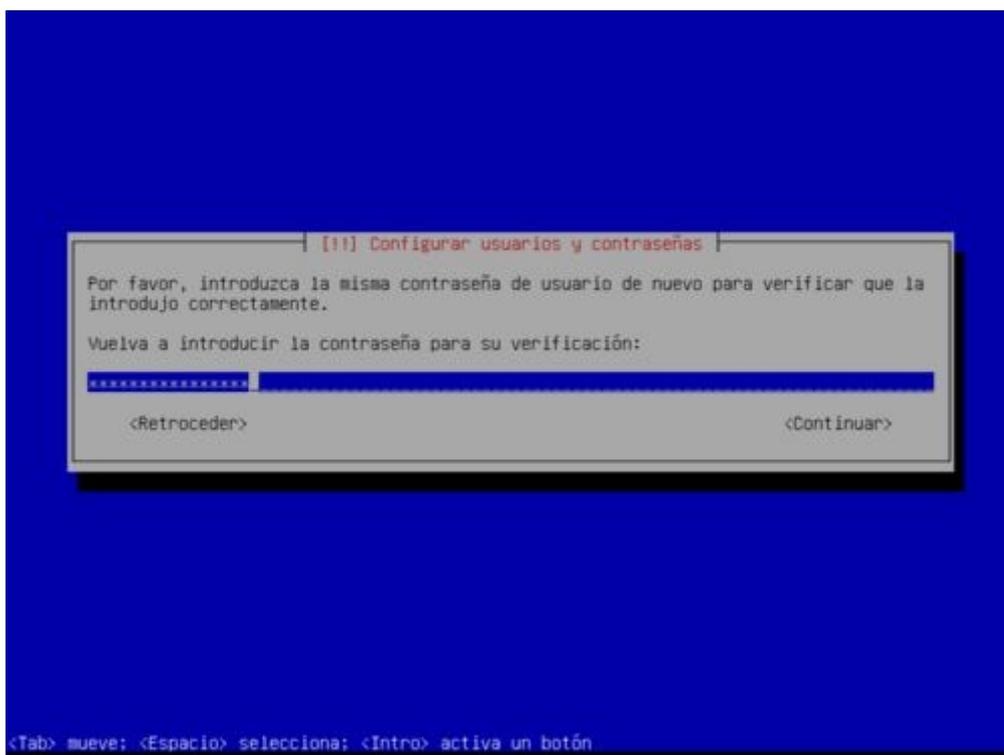
Introduce tu nombre completo tal y como nos aconsejan en la instalación (*Si quieres*) o un seudónimo como es mi caso, continuamos pulsando la tecla (*Enter*) ...



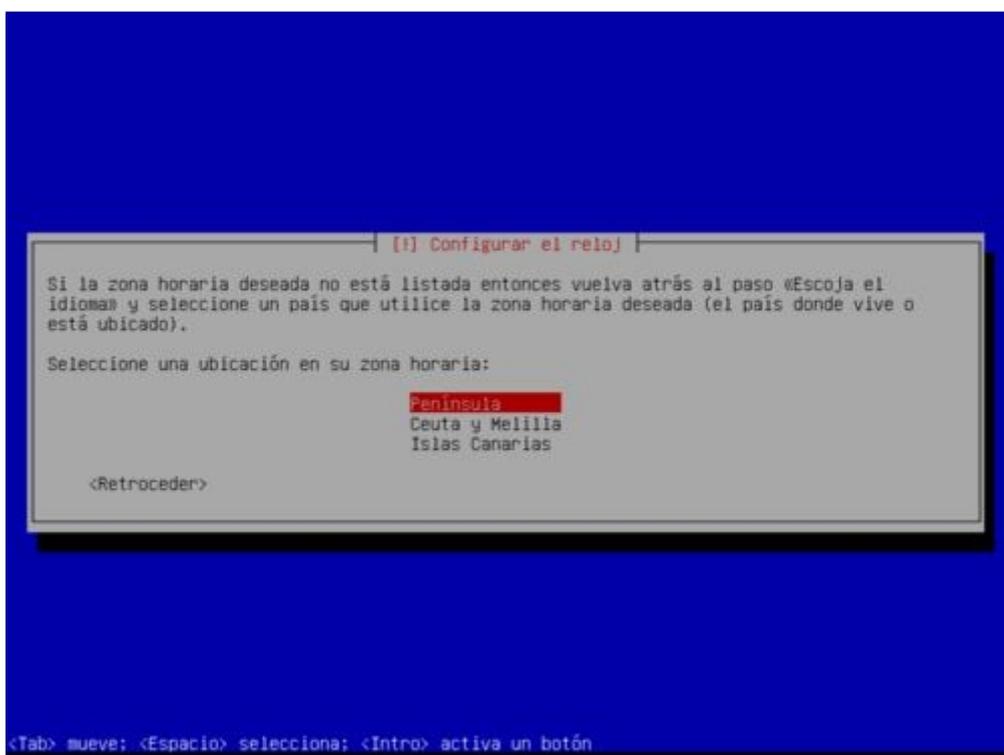
Introducimos un nombre de usuario con el que nos *identificaremos* en la maquina y con el que se creará tu directorio personal en la (*/home*), continuamos pulsando la tecla (*Enter*) ...



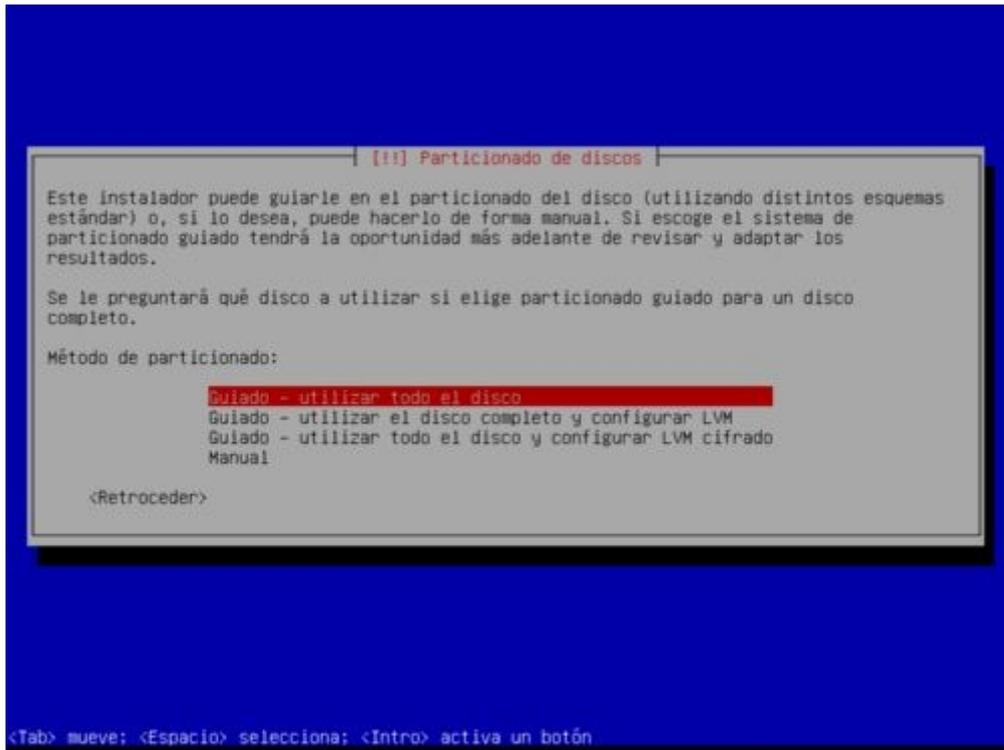
Introducimos una *contraseña* para el usuario que acabamos de crear y pulsamos la tecla (*Enter*) ...



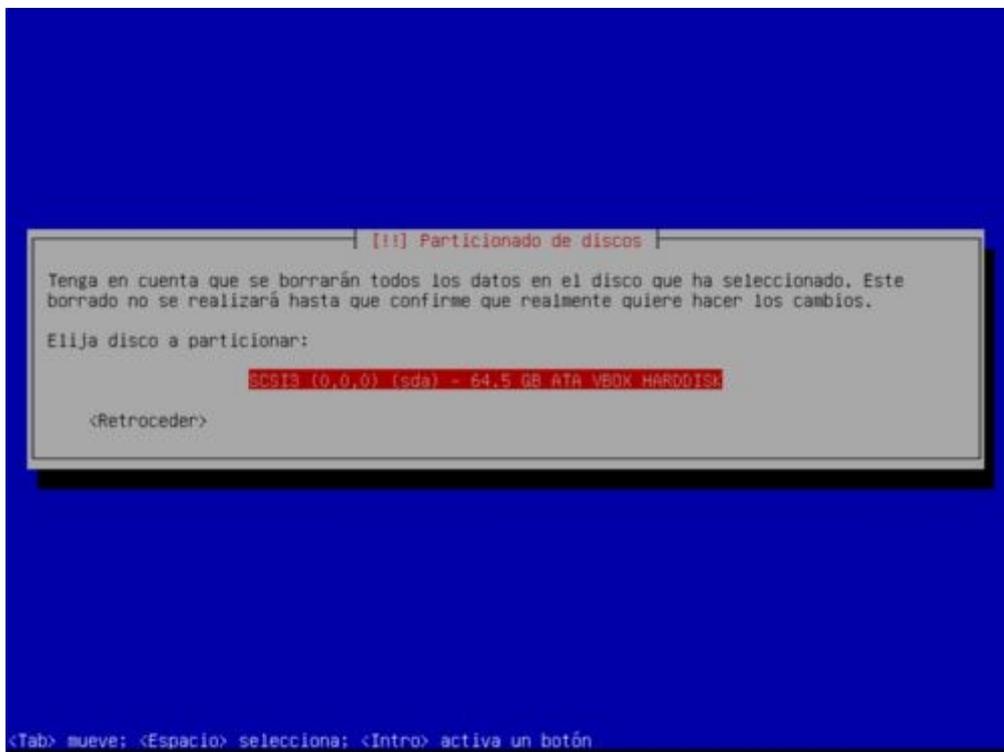
Introducimos de nuevo la misma *contraseña* y pulsamos la tecla (*Enter*) ...



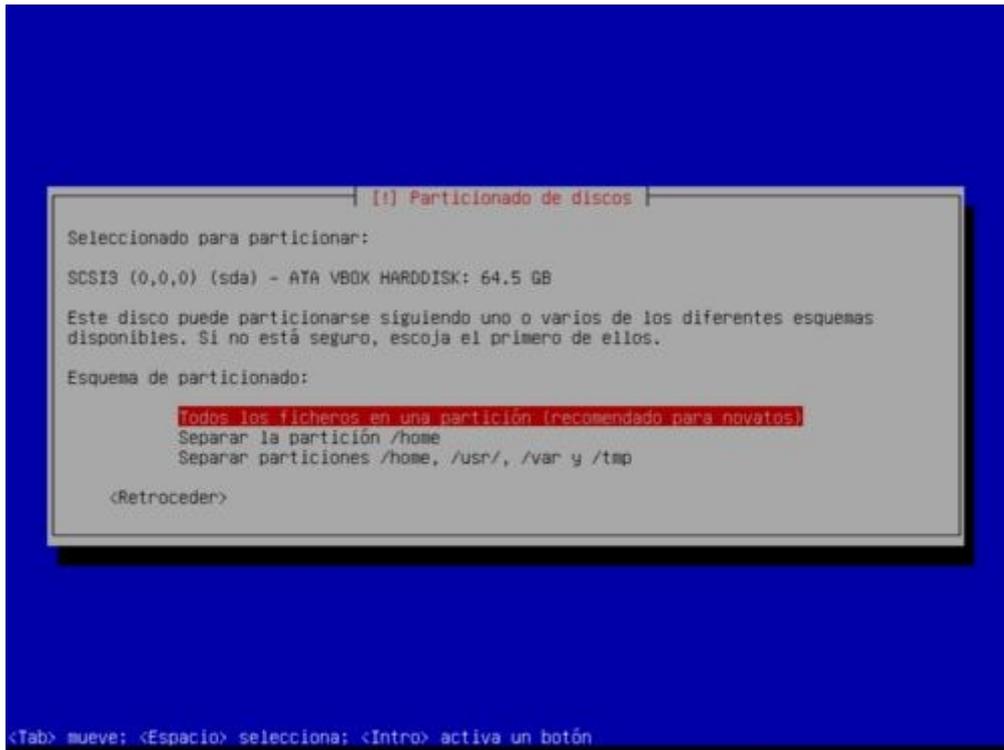
Introducimos nuestra *zona horaria* y pulsamos la tecla (*Enter*) ...



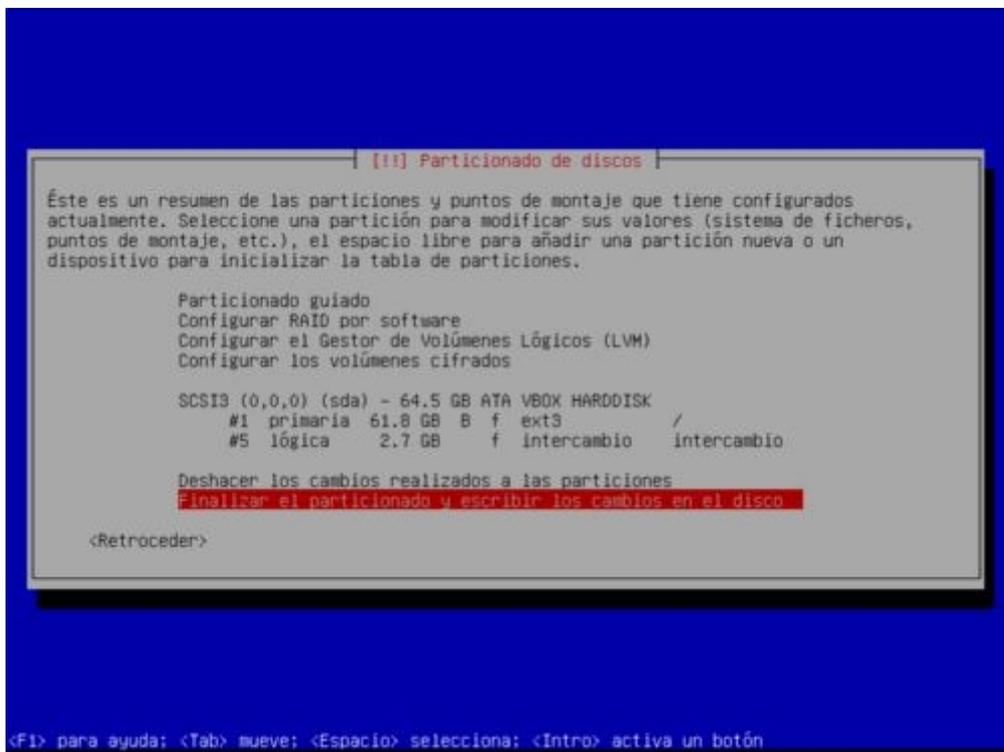
Seleccionamos la opción (*Utilizar todo el disco*) y pulsamos la tecla (*Enter*) ...



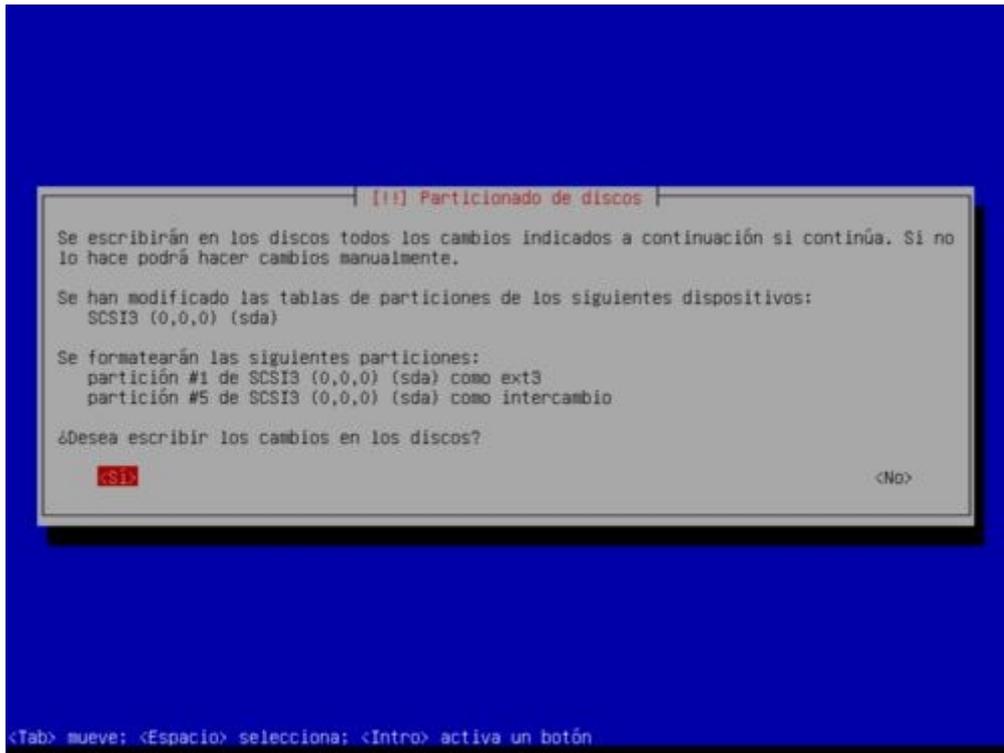
Seleccionamos el *disco duro* donde queremos instalarlo y pulsamos la tecla (*Enter*) ...



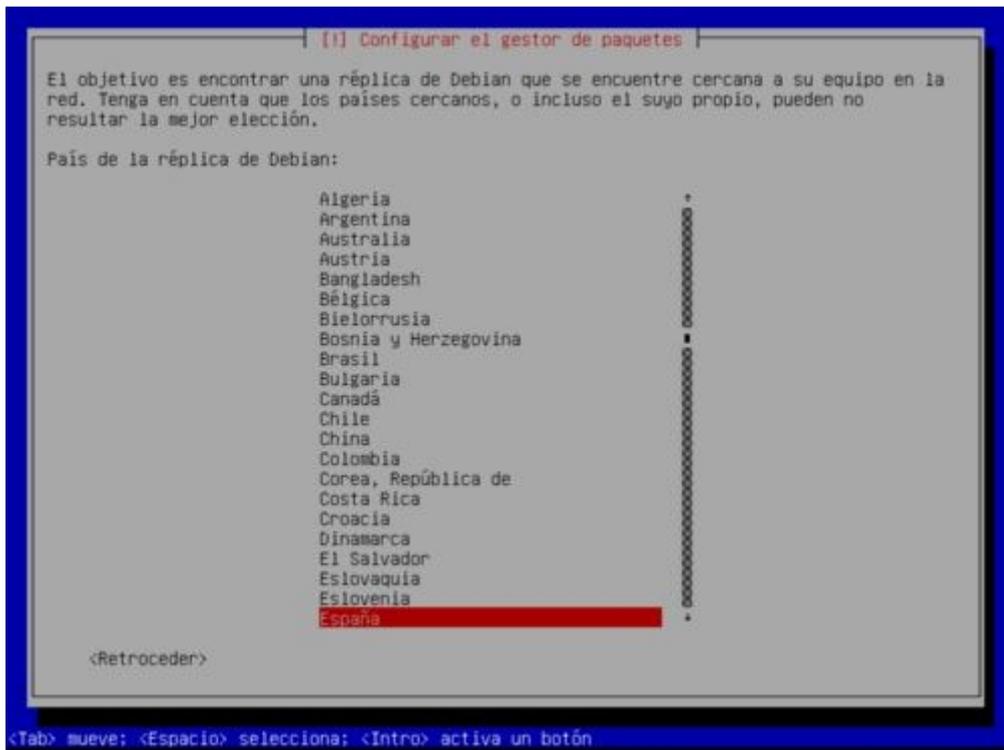
Seleccionamos (*Todos los ficheros en una partición*) y pulsamos la tecla (*Enter*) ...



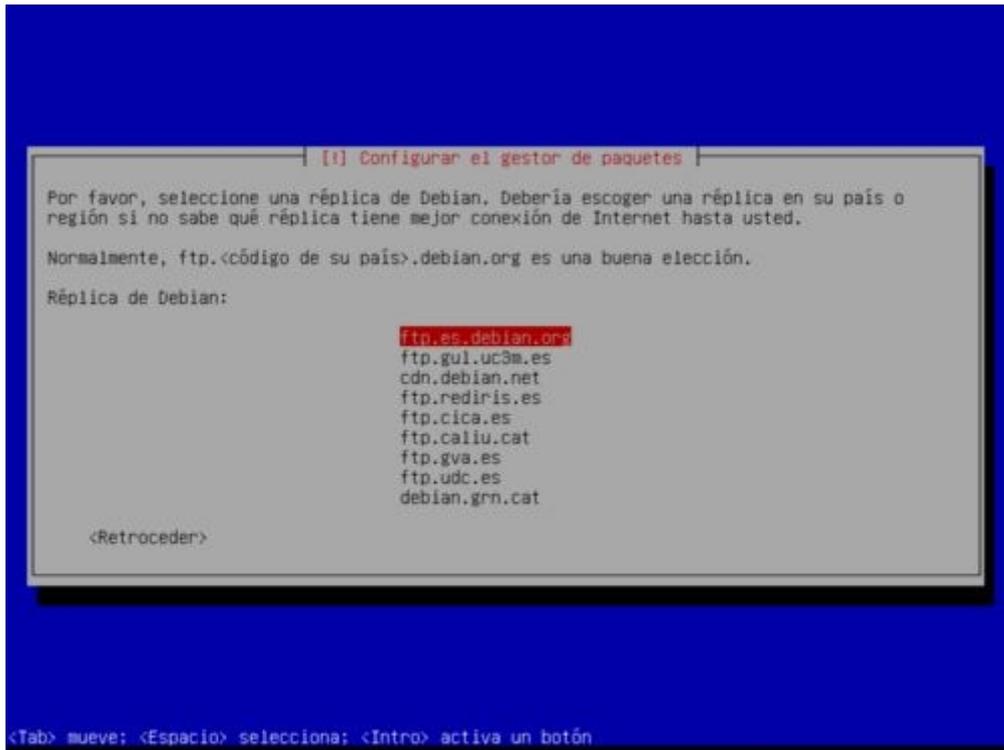
Como podemos ver se han creado dos particiones automáticamente, la (/) y la (*SWAP*), si vemos que está todo correcto selecciona la opción (*Finalizar el particionamiento y escribir los cambios en el disco*) y pulsamos la tecla (*Enter*) ...



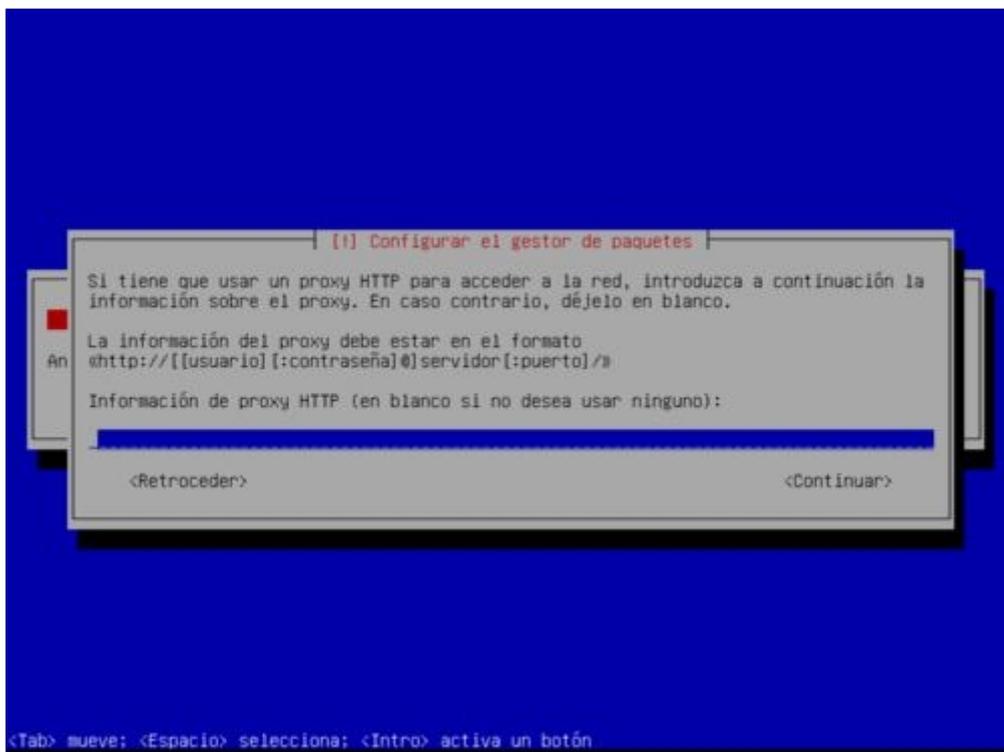
Como medida de seguridad nos pregunta de nuevo si estamos de acuerdo con las particiones que se van a montar, si está todo correcto seleccionamos la opción (*Si*) y pulsamos la tecla (*Enter*) ...



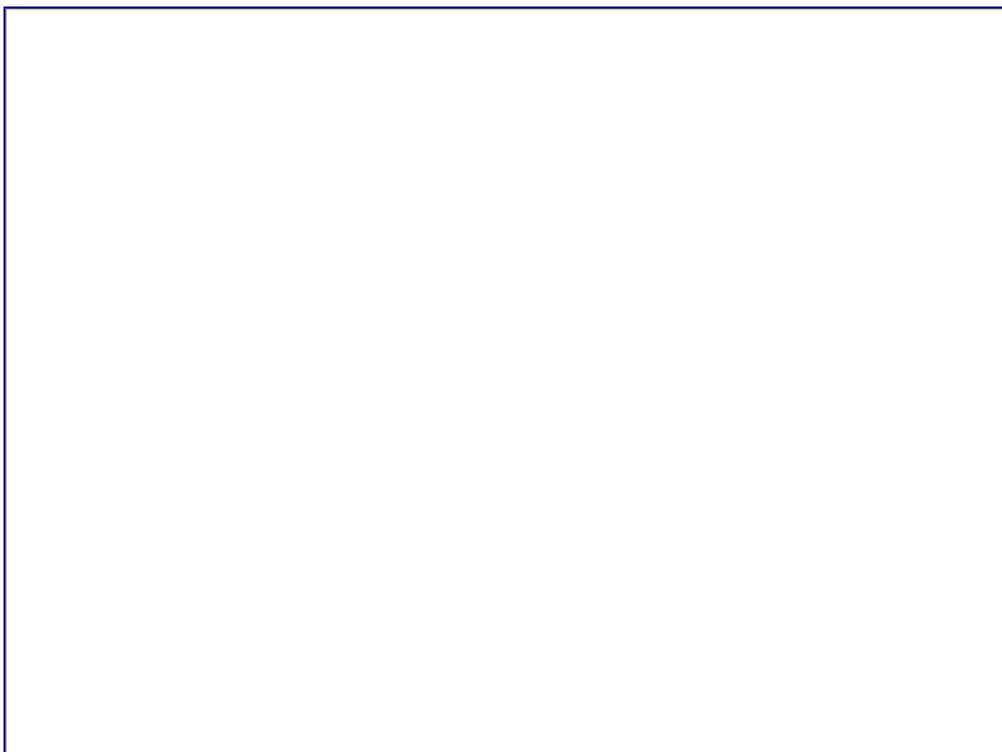
Seleccionamos nuestro país o el mas cercano a nosotros y pulsamos la tecla (*Enter*) ...



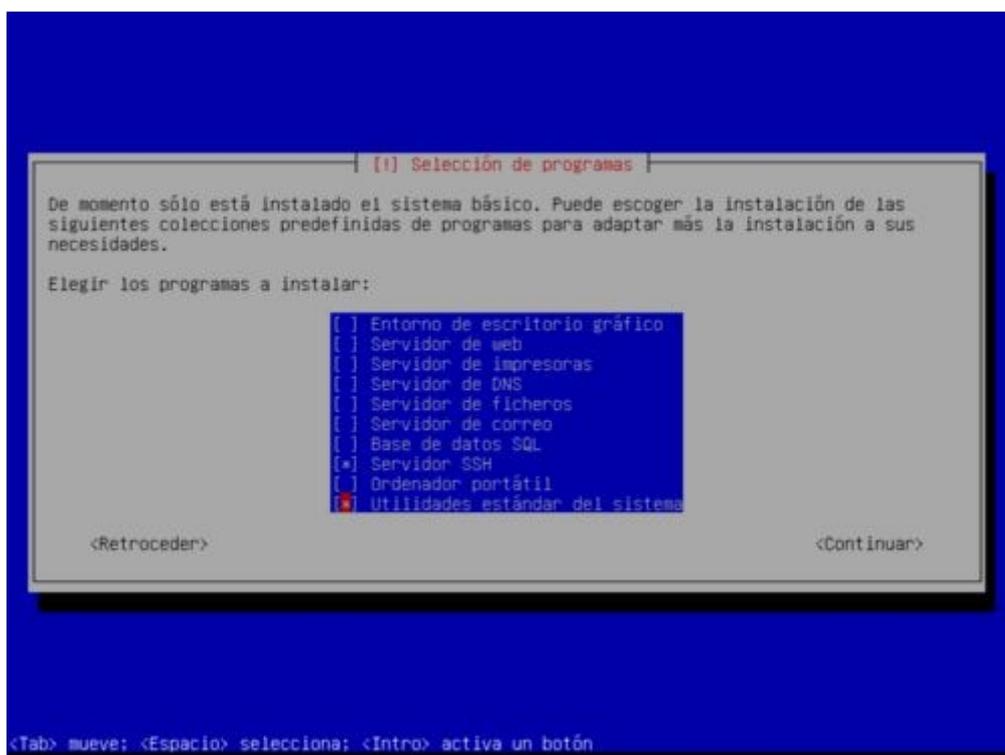
Elejimos la dirección mas cercana o la que queramos (*funcionará de todos modos*) y pulsamos la tecla (*Enter*) ...



Introducimos la dirección del Proxy si es que estamos detras de uno, de no ser así dejamos el espacio vacío y pulsamos la tecla (*Enter*) ...

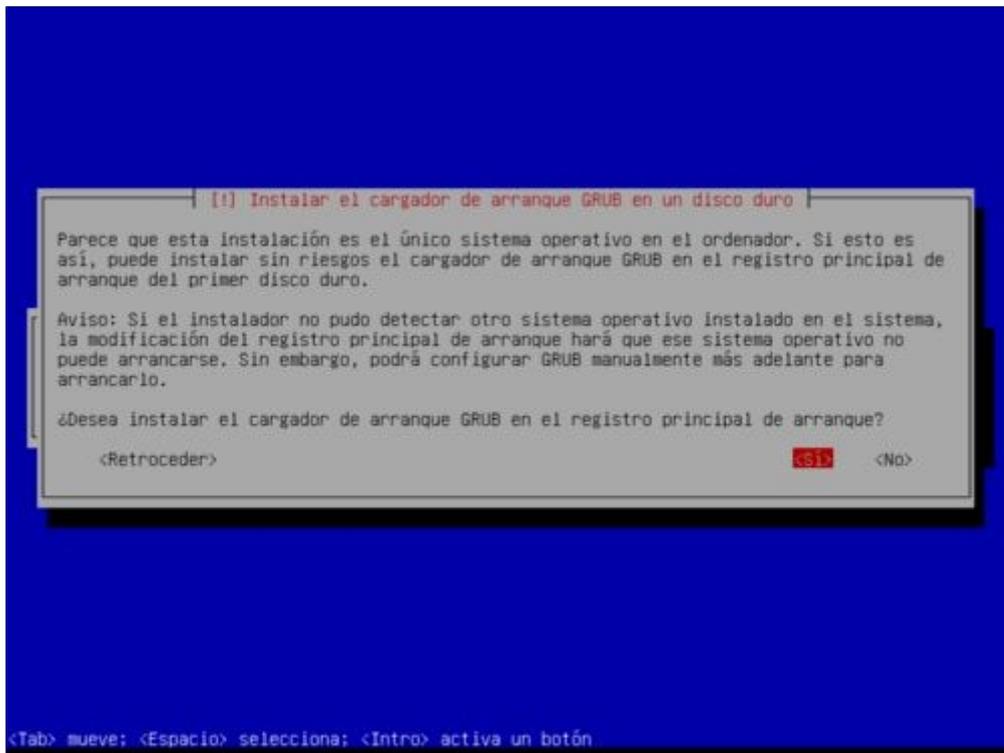


Esta opción os la dejo a vuestra voluntad ya que cada uno hace lo que quiere aquí, pulsamos la tecla (*Enter*) y continuamos ...

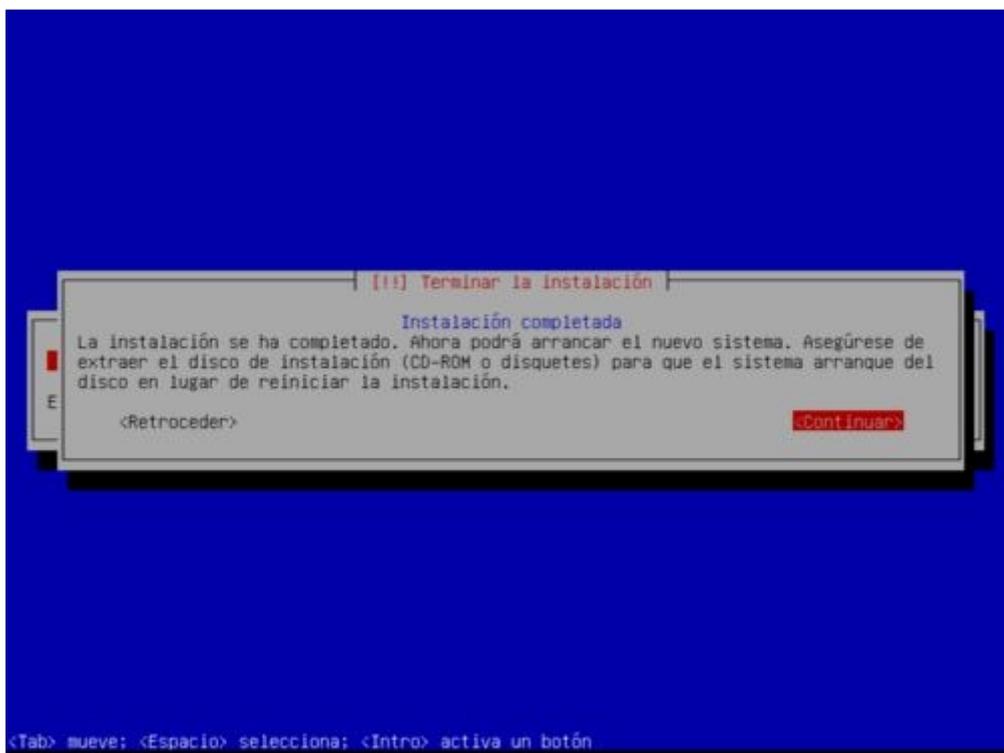


Desmarcamos la opción (*Entorno de escritorio gráfico*) con la barra espaciadora. Tan solo tenemos que seleccionar (*Servidor SSH*) y (*Utilidades estándar del sistema*), una vez marcadas

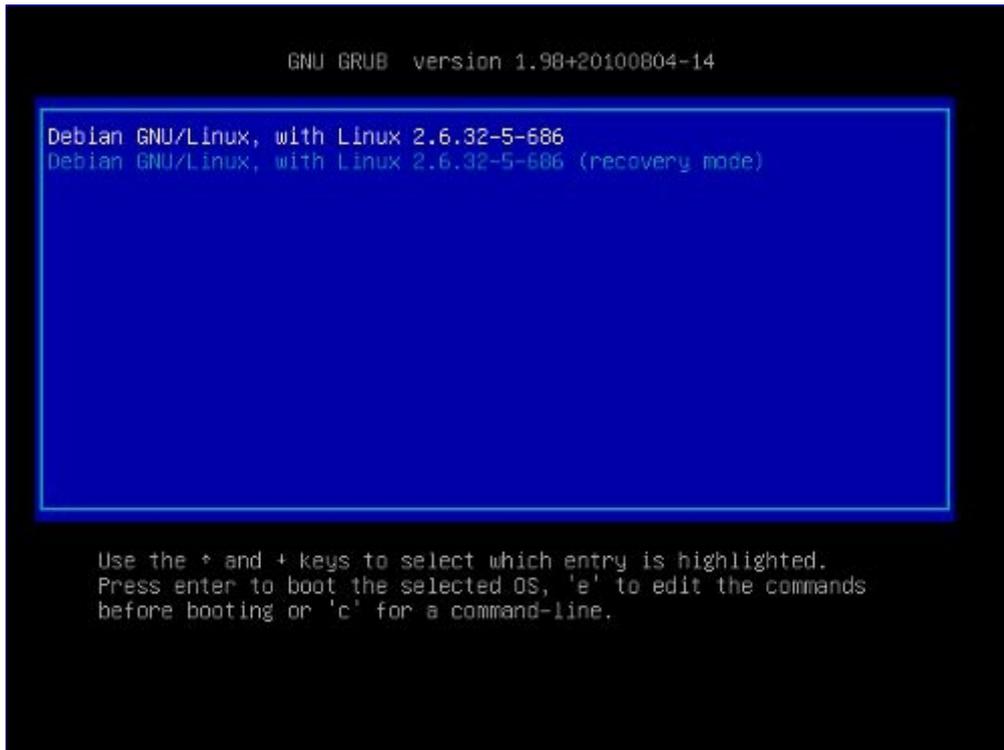
pulsamos la tecla (*Enter*) ...



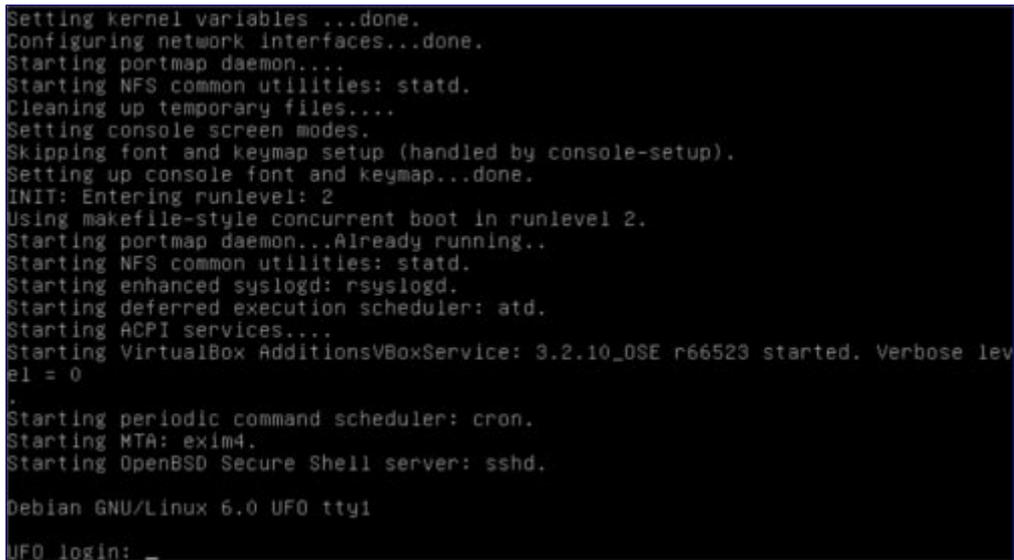
Seleccionamos la opción (*Si*) y pulsamos la tecla (*Enter*) ...



Y después de 25 pasos que acabamos de dar pulsamos por ultima vez la tecla (*Enter*) y el sistema se reiniciará. *Extraemos el CD de instalación y listo ...*



Como vemos ya tenemos al cargador de sistemas operativos (**GRUB**) preparado para la acción, seleccionamos la primera opción y pulsamos la tecla (*Enter*) ...



Ahora nos identificamos por ejemplo como (*root*) e introducimos la contraseña que le dimos durante la instalación y pulsamos la tecla (*Enter*) ...

```
Starting NFS common utilities: statd.
Starting portmap daemon...Already running..
Starting enhanced syslogd: rsyslogd.
Starting VirtualBox AdditionsVBoxService: 3.2.10_OSE r66523 started. Verbose level = 0
.
Starting ACPI services....
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.
Starting MTA: exim4.
Starting OpenBSD Secure Shell server: sshd.

Debian GNU/Linux 6.0 UFO tty1

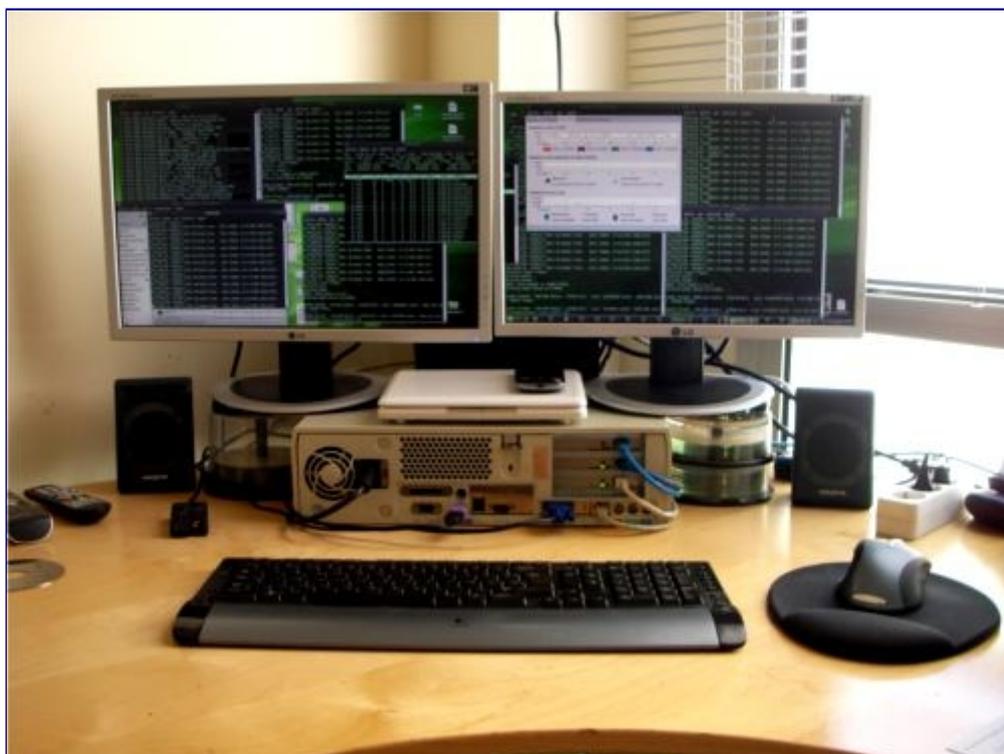
UFO login: root
Password:
Linux UFO 2.6.32-5-686 #1 SMP Fri Sep 9 20:51:05 UTC 2011 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@UFO:~#
```

En estos momentos nos encontramos con el sistema puro y duro en la línea de comandos en la que tendremos que aprender a defendernos si queremos sobrevivir.

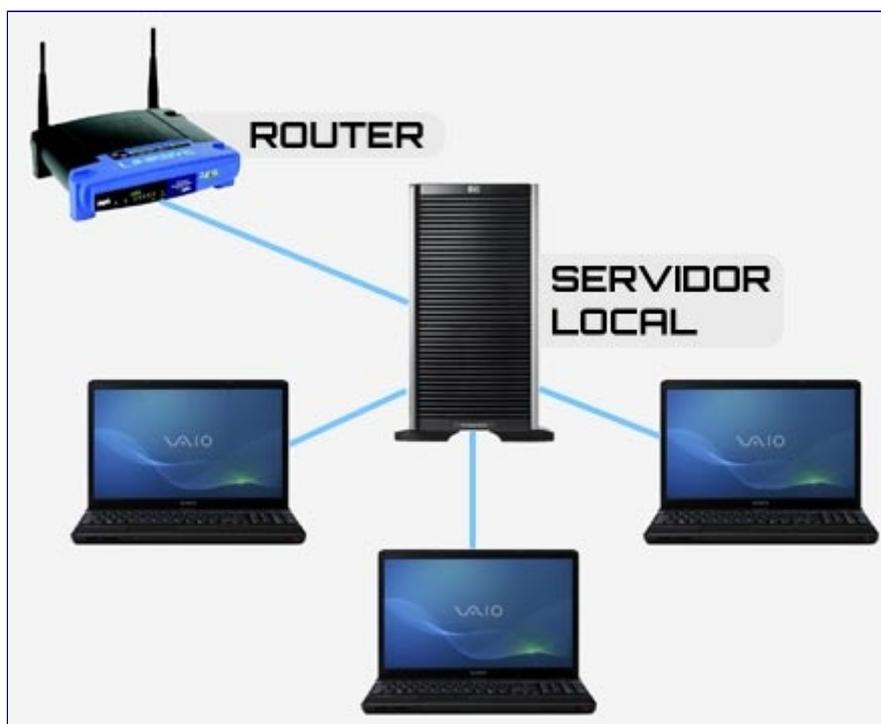
Antes de introducir el comando (*halt*) y apagar la maquina os recomendaría que probarais el comando (*top*) ...



Con este comando podemos ver a tiempo real como se encuentra nuestro servidor. Cosas como el tiempo uptime, hora local, usuarios conectados, carga de la maquina (load average) en el ultimo minuto, cinco y quince, tareas ejecutadas, en espera, paradas y zombies además de consumo de CPU/s, memoria ram usada y libre contando con la de intercambio (swap) y procesos en ejecución. Todo a tiempo real como si fuéramos médicos y viéramos que tal esta nuestro paciente, el consumo de recursos puede ser un gran problema para un servidor, aquí nos daremos cuenta rápido de que pasa. Para salir de la aplicación (*top*) tan solo tienes que pulsar las teclas (*Ctrl + z*) o también pulsando la tecla (*q*). Ya tenemos la **instalación de Debian** completada con los paquetes mínimos incluido el **servidor SSH** para que podamos conectarnos a el remotamente. Hasta aquí llega el primer capítulo del proyecto.

Vol 2 - Como montar un Switch de Red

Bueno amigos, continuamos con el proyecto ([Viejos ordenadores que hacen grandes cosas](#)). Como decíamos en el artículo vamos a incluir unas cuantas **tarjetas de red** a nuestro querido **#Servidor Local**. Agregándole varias tarjetas de red podemos montarnos un **Switch de red** ...



La idea es bien sencilla, nuestro **Router** lo conectamos a nuestro servidor por una de las **tarjetas de red** y los otros ordenadores que tengamos tan solo tendremos que conectarlos a las demás tarjetas libres y tendremos acceso al servidor, claro está también hacia Internet.

En mi caso el servidor tiene una **tarjeta de red** integrada y tres extras que le he instalado yo, con lo que tenemos cuatro tarjetas de red para montar el **Switch de red**.

No es necesario tener cuatro tarjetas como es mi caso si no que tan solo necesitamos que nuestro servidor tenga dos o mas tarjetas de red. Durante el tutorial agregaremos tanto dos como las que sean.

Si estas preparado tan solo tienes que arrancar tu **#Servidor Local** e identificarte como súper usuario (*root*).

```
Starting NFS common utilities: statd.
Starting portmap daemon...Already running..
Starting enhanced syslogd: rsyslogd.
Starting VirtualBox AdditionsVBoxService: 3.2.10_OSE r66523 started. Verbose level = 0
.
Starting ACPI services...
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.
Starting MTA: exim4.
Starting OpenBSD Secure Shell server: sshd.

Debian GNU/Linux 6.0 UFO tty1

UFO login: root
Password:
Linux UFO 2.6.32-5-686 #1 SMP Fri Sep 9 20:51:05 UTC 2011 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@UFO:~#
```

Lo primero que haremos una vez iniciada la sesión será **configurar las tarjetas de red** para que en una conectemos el **Router** y en el resto los ordenadores que queramos.

Para configurar esto montaremos un **punto de red** o también llamado **Bridge**. El modo *Bridge* puede hacer que las 4 tarjetas tengan la misma IP actuando todas como una y no de manera independiente.

Para hacer funcionar el invento necesitamos el paquete (*bridge-utils*) y lo instalaremos tecleando lo siguiente ...

```
root@UFO:~# apt-get install bridge-utils
```

apt-get install bridge-utils

Una vez instalado hacer una copia del fichero (*interfaces*) porque vamos a modificarlo y siempre nos puede venir bien en caso de fallo ...

```
root@UFO:~# cp /etc/network/interfaces /etc/network/interfacesOLD
```

cp /etc/network/interfaces /etc/network/interfacesOLD

Seguidamente editamos el fichero de configuración de la red con ...

```
root@UFO:~# vi /etc/network/interfaces
```

vi /etc/network/interfaces

Borramos todo el contenido pulsando la tecla (*ESC*) y seguidamente (*:d 100*). Una vez limpio introducimos lo siguiente basándonos en los datos de vuestra red y en vuestras tarjetas de red ...

```
root@UFO:~#
```

```
auto lo
iface lo inet loopback
auto br0
iface br0 inet static
address 192.168.1.240 # IP estática para nuestro server
network 192.168.1.0
netmask 255.255.255.0
broadcast 192.168.1.255
gateway 192.168.1.254 # IP de nuestro Router
bridge_ports eth0 eth1 eth2 eth3 # Estos son los identificativos de las tarjetas
que tengas
bridge_maxwait 0
```

Una vez introducidos estos datos salvaremos el fichero y saldremos del editor (*vi*) pulsando la tecla (*ESC*) y seguidamente (*:wq!*).

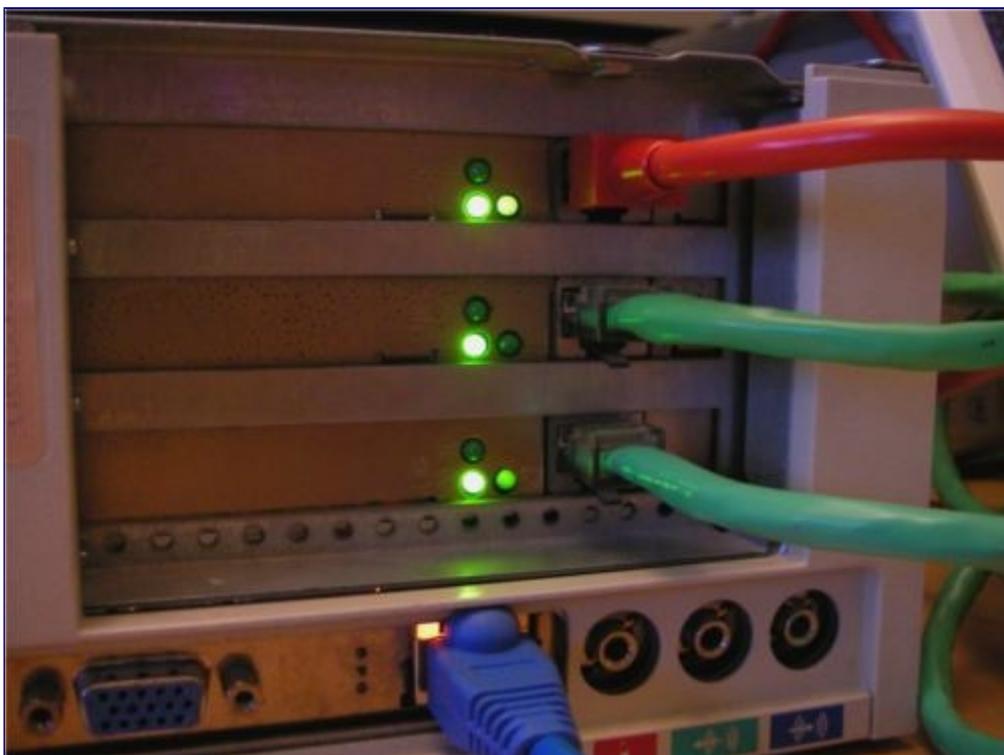
Ahora reiniciamos las tarjetas de red con ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

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

Si tecleamos el comando (*ifconfig*) veremos que se muestran todas las tarjetas de red pero solo tiene una de ellas una IP, la identificada como (*br0*) que en realidad es el puente de red.

Ahora que nuestras **tarjetas de red** están listas y en una de ellas tenemos el **Router** conectado vamos a probar el invento conectando otros ordenadores a las tarjetas de red que quedan libres ...



Ahora desde otro ordenador conectado a nuestra red vamos a acceder al servidor desde una *Terminal* tecleando lo siguiente ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

```
ssh root@IP.DE.TU.SERVER
```

Introducimos la contraseña del súper usuario (*root*) y ya estamos dentro !

Ahora si desde este ordenador cliente arrancamos el navegador veremos que tenemos conexión Internet sin problemas. Ya podemos deshacernos de nuestro antiguo **switch de red** y de sus cables por supuesto 😊

Vol 3 - Como montar un servidor DNS

Continuando con el proyecto sobre ([Viejos ordenadores que hacen grandes cosas](#)) vamos a equipar nuestro #Servidor Local con un **servidor DNS Local**. Me refiero con **servidor DNS Local** a que nuestro servidor se dedicará a resolver nombres de dominio y pasaremos de el que nos brinda nuestro **ISP**.

Ahora si un día ves que tienes conexión a internet y no te resuelve el **DNS** no tendrás que llamar al servicio técnico simplemente porque el servidor lo tendrás tu y sabrás que pasa ...



Esta no es la única ventaja que tiene el tener un **servidor DNS** en casa, si no que el tiempo de acceso al mismo es mucho mas rápido y se nota bastante en la navegación porque tarda menos tiempo en resolver y responder.

Para los que tenéis **ADSL** de 20 Mb no se si notareis la diferencia cuando estéis navegando por la red pero para los que tenemos una **ADSL** de 3 Mb como es mi caso se nota mucho. Si estáis interesados en agregarle esta función a vuestro servidor os acompaño en el camino, arrancad vuestras maquinas, identificaros como súper usuario (*root*) y comenzamos ...

```
Starting NFS common utilities: statd.
Starting portmap daemon...Already running..
Starting enhanced syslogd: rsyslogd.
Starting VirtualBox AdditionsVBoxService: 3.2.10_OSE r66523 started. Verbose level = 0
.
Starting ACPI services...
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.
Starting MTA: exim4.
Starting OpenBSD Secure Shell server: sshd.

Debian GNU/Linux 6.0 UFO tty1

UFO login: root
Password:
Linux UFO 2.6.32-5-686 #1 SMP Fri Sep 9 20:51:05 UTC 2011 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@UFO:~#
```

Antes de nada vamos a hacer unas copias de seguridad de los archivos que vamos a modificar, en caso de error siempre podremos ir a la configuración original ...

```
00001100010001001001001 CODIGO 100010010000010100010
```

```
cp /etc/resolv.conf /etc/resolv.confOLD
cp /etc/bind/named.conf.options /etc/bind/named.conf.optionsOLD
```

Para que nuestro servidor pueda resolver nombres y poder usarlo en vez de el que te asigna tu *proveedor de internet* (**ISP**) necesitamos hacer unas cuantas configuraciones, para comenzar vamos a instalar el servidor DNS BIND tecleando lo siguiente ...

```
00001100010001001001001 CODIGO 100010010000010100010
```

```
apt-get install bind9
```

A continuación vamos a editar el fichero de **configuración DNS** (*resolv.conf*) con ...

```
00001100010001001001001 CODIGO 100010010000010100010
```

```
vi /etc/resolv.conf
```

En mi caso la URL que apunta hacia mi servidor es <http://hsnet.sytes.net> y la **IP** de mi **Router** es *192.168.1.254*, introduce tus datos ...

```
00001100010001001001001 CODIGO 100010010000010100010
```

```
domain hsnet.sytes.net
search hsnet.sytes.net
nameserver 127.0.0.1
nameserver 192.168.1.254
```

Una vez modificado pulsamos la tecla (*ESC*) y seguidamente (*:wq!*).

A continuación editaremos el fichero de configuración (*named.conf.options*) para decirle que resuelva de manera local con lo siguiente ...



```
vi /etc/bind/named.conf.options
```

Buscamos las líneas ...



```
// forwarders {  
    //     0.0.0.0;  
    // };
```

Las descomentamos y las modificamos con nuestros datos (*IPs -> localhost y Router*)...



```
forwarders {  
    127.0.0.1;  
    192.168.1.254;  
};
```

Salimos del fichero pulsando la tecla (*ESC*) y seguidamente (*:wq!*).

Reiniciamos el **servidor DNS Bind** para que coja los cambios con ...



```
/etc/init.d/bind9 restart
```

Ahora nuestro **servidor** auto resolverá direcciones y podremos usarlo en todos los ordenadores que tengamos conectados a nuestra red.

Para usarlo con los ordenadores cliente que teneis conectados al **#Servidor Local** tan solo tenéis que ir hacia las configuraciones de la red e introducir como **servidor DNS** la **IP** de vuestro **servidor**.

La primera vez que carguemos una web tardará mas o menos como el de vuestro **ISP** pero a partir de ahí la almacenará en su cache para ofrecerla aun mas deprisa. *Tiempo de respuesta cero coma ... (Bonito, Bonito)*

Vol 4 - Como montar un servidor Web

Si hay un **hosting economico** este va a ser el nuestro y como no, en este proyecto llamado ([Viejos ordenadores que hacen grandes cosas](#)) estamos montando un servidor que estará conectado a la red las 24 horas y lo suyo sería montar en el nuestras paginas web o blogs. Dependiendo del ancho de banda de tu **ADSL** podrás ofrecer mas paginas o menos. Si tu ancho de banda es muy pequeño es posible que al entrar 8 o 9 personas a leer tu web no tenga suficiente y no pueda cargar sin mas. Seguramente el ancho de banda de tu **ADSL** sera un impedimento mayor frente a tu **servidor**. Si recordamos montamos un **punte de red** en nuestro servidor con unas tarjetas de red de 10/100 Mbps como mínimo con lo que podría aguantar un máximo de 10 Mbps de subida. En la red local será raro que se quede corto de ancho de banda pero hacia internet podremos experimentar problemas cuando tengamos mucho trafico. *No os preocupéis que si os vais quedando cortos de ancho de banda por culpa de vuestra ADSL os daréis cuenta rápido ...*



Aun así vamos a instalar y configurar un servidor web tipo **LAMP**, **Linux** + **Apache** + **MySQL** + **PHP**, en el podremos montar un Blog tipo WordPress, una tienda online, archivos para descargar o lo que queramos ofrecer vía web.

Para comenzar la instalación no está demás que digamos que hay que identificarse en nuestro **#Servidor Local** como súper usuario (*root*) ...

```
Starting NFS common utilities: statd.
Starting portmap daemon...Already running..
Starting enhanced syslogd: rsyslogd.
Starting VirtualBox AdditionsVBoxService: 3.2.10_OSE r66523 started. Verbose level = 0
.
Starting ACPI services...
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.
Starting MTA: exim4.
Starting OpenBSD Secure Shell server: sshd.

Debian GNU/Linux 6.0 UFO tty1

UFO login: root
Password:
Linux UFO 2.6.32-5-686 #1 SMP Fri Sep 9 20:51:05 UTC 2011 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@UFO:~#
```

Instalación y configuración de Apache Web Server

```
root@UFO:~# apt-get install apache2 apache2-mpm-prefork
```

apt-get install apache2 apache2-mpm-prefork

Una vez instalado el **servidor web Apache** vamos a crear un directorio donde copiaremos todo lo que queramos servir vía web como por ejemplo ...

```
root@UFO:~# mkdir /home/usuario/web
```

mkdir /home/usuario/web

Ahora editaremos el fichero (*default*) de **Apache** para indicarle que lea el contenido web desde el directorio que creamos anteriormente ...

```
root@UFO:~# vi /etc/apache2/sites-available/default
```

vi /etc/apache2/sites-available/default

Buscamos la línea ...

```
root@UFO:~# DocumentRoot /var/www/
```

DocumentRoot /var/www/

Y sustituimos la ruta por la del directorio que creamos anteriormente ...

```
root@UFO:~# DocumentRoot /home/usuario/web/
```

DocumentRoot /home/usuario/web/

Un poco mas abajo encontramos la linea ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

Directory /var/www/

Y la sustituimos por ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

Directory /home/usuario/web/

Seguidamente y para finalizar buscamos la linea ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

AllowOverride None

Y la modificamos por ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

AllowOverride all

Salimos del editor pulsando la tecla (*ESC*) y seguidamente (*:wq!*)

Recargamos el **servidor web Apache** con ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

/etc/init.d/apache2 reload

Ahora si introducimos la IP de nuestro servidor desde un navegador veremos el directorio (*web*) ...

Index of /

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
<hr/>			
 hsnet/	10-Sep-2011 02:10	-	
<hr/>			
Apache/2.2.9 (Debian) Server at hsnet.sytes.net Port 80			

Seguridad por oscuridad

Como podemos ver en la parte inferior **Apache** nos ofrece por defecto su versión y el sistema operativo que lo acoge. No pasa nada porque se vean estos datos pero estaría mejor si no se viesen. Seguridad por oscuridad es una técnica que trata de revelar los datos mínimos respecto a los

programas que un **servidor** tiene instalados. No es lo mismo que el atacante sepa cosas del objetivo, que no sepa nada.

Para dejar de mostrar estos datos editaremos el fichero de configuración de **Apache** con ...



```
vi /etc/apache2/apache2.conf
```

Bajaremos al final del fichero e insertaremos estas dos líneas ...



```
ServerSignature Off  
ServerTokens Prod
```

Afinando Apache ...

Ya que estamos vamos a hacer unas cuantas modificaciones para que nuestro servidor web apache pueda aguantar mas peticiones y aligere el trafico, la carga ...

Buscamos la línea ...



```
Timeout 300
```

Y la modificamos por ...



```
Timeout 15
```

Buscamos la línea



```
KeepAliveTimeout 15
```

Y la modificamos por ...



```
KeepAliveTimeout 3
```

Buscamos la línea ...



```
MaxRequestsPerChild 0
```

Y la modificamos por ...

```
0000110001000100100101 CODIGO 10001001000001010100010
```

```
MaxRequestsPerChild 1000
```

Una vez hechos estos cambios salimos del editor y guardamos pulsando la tecla (*ESC*) y seguidamente (*:wq!*)

Para finalizar vamos a darle soporte para **PHP5** instalando los siguientes paquetes ...

```
0000110001000100100101 CODIGO 10001001000001010100010
```

```
apt-get install php5 libapache2-mod-php5 php5-mysql php5-curl  
php5-gd php5-idn php-pear php5-imagick php5-imap php5-mcrypt  
php5-memcache php5-ming php5-ps php5-pspell php5-recode php5-snmp  
php5-sqlite php5-tidy php5-xmllrpc php5-xsl
```

Recargamos el **servidor web Apache** para que acepte los cambios con ...

```
0000110001000100100101 CODIGO 10001001000001010100010
```

```
/etc/init.d/apache2 restart
```

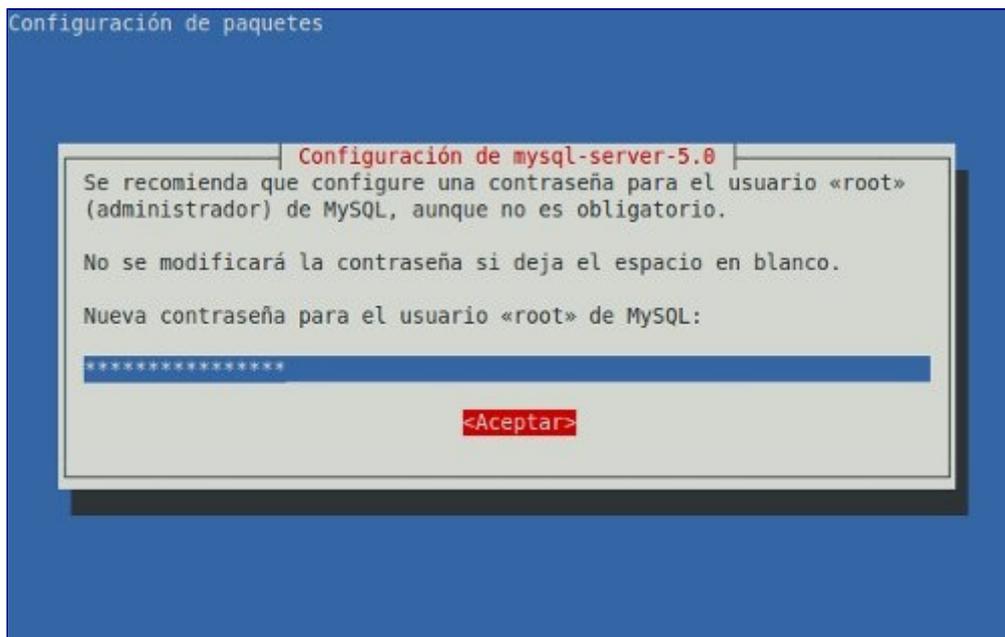
Instalación y configuración MySQL + PhpMyAdmin ...

Seguidamente instalaremos el **servidor** para las **bases de datos MySQL** además de **phpMyAdmin** para poder administrar nuestras bases de datos vía web ...

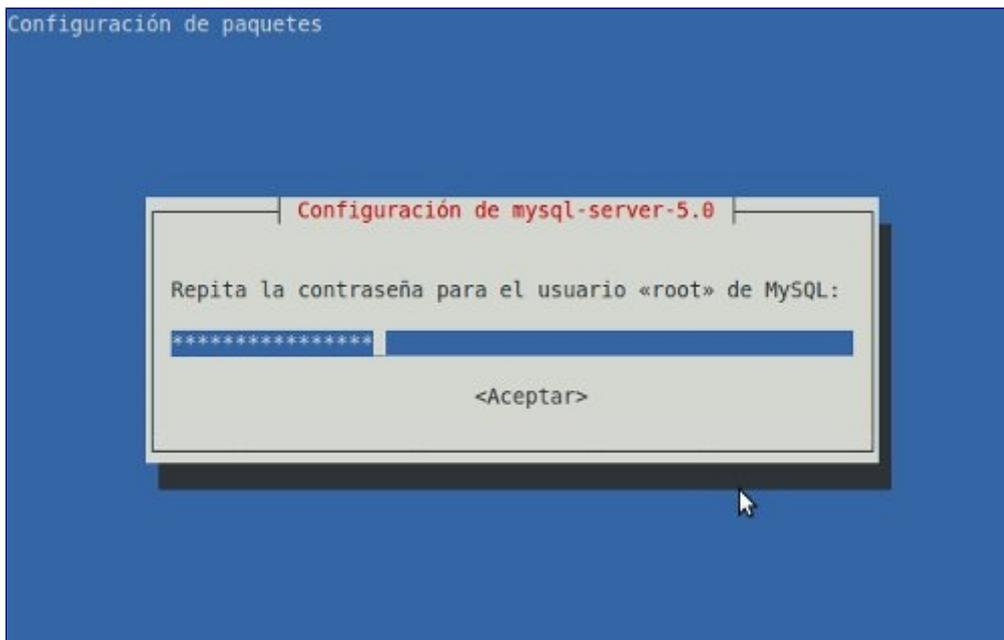
```
0000110001000100100101 CODIGO 10001001000001010100010
```

```
apt-get install mysql-server mysql-client phpmyadmin
```

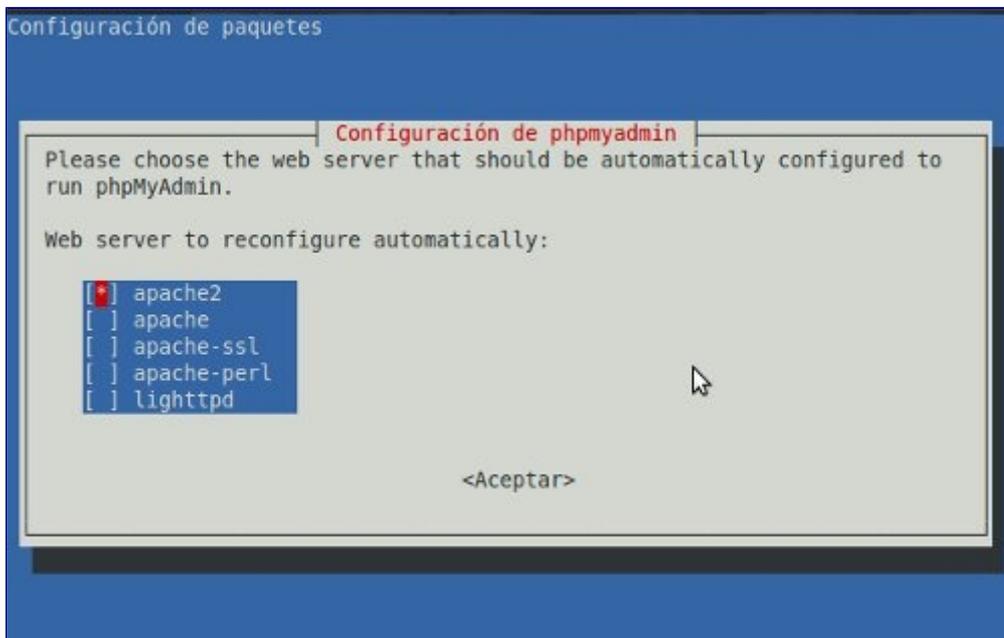
Durante la instalación nos encontramos con lo siguiente ...



Introducimos una contraseña para el usuario (*root*) que sera el administrador de las bases de datos **MySQL** y pulsamos la tecla (*Enter*)...



Introducimos de nuevo la misma contraseña y pulsamos la tecla (*Enter*) ...



Marcamos la opción (*Apache2*), pulsamos la tecla (*Enter*) y listo.

Para acceder al administrador de las bases de datos MySQL, ve hacia otro ordenador, arranca el navegador e introduce <http://IPDETUSERVER/phpmyadmin> y veremos lo siguiente ...



Introduce (*root*) como nombre de usuario, la contraseña que le diste anteriormente y pulsa la tecla (*Enter*) ...



Como podemos ver ya estamos dentro del administrador de las bases de datos MySQL (**phpMyAdmin**), desde aquí podemos agregar, reparar, optimizar, editar, modificar, eliminar y todo lo que necesitemos hacerles a nuestras bases de datos de una manera sencilla y sobre todo web.

Ya tenemos nuestro servidor Web integrado en nuestro **#Servidor Local**, recordemos que según este tutorial el directorio raíz del servidor web es (*/home/usuario/web/*) y para acceder desde el navegador (*http://IP.DE.TU.SERVER*). Todo lo que incluyamos en ese directorio se vera reflejado vía web. Para acceder al administrador de las **bases de datos MySQL** es (*http://IP.DE.TU.SERVER/phpmyadmin*).

En el siguiente volumen montaremos unos cuantos discos duros extras y los compartiremos con nuestra red de ordenadores con *GNU/Linux* vía *NFS*.

Vol 5 - Como montar un servidor NFS

Otra de las cosas que nuestro **#Servidor Local** puede hacer por y para nosotros es convertirse en un almacén de datos con el que acceder a el sea tan sencillo como hacer click en un icono de acceso directo en el escritorio de todos los ordenadores cliente que tengamos conectados a el. Aunque a día de hoy existan *mini servidores NAS, discos duros externos y demás soportes de almacenaje* una vez mas nuestro proyecto ([Viejos ordenadores que hacen grandes cosas](#)) puede sernos útil simplemente porque ha llegado el momento de usar aquellos discos duros que tienes aparcados en un cajón de tu habitación para convertirlos en tu almacén de datos. Una vez instalado correctamente podrás leer tu musica, vídeos, ficheros y demás directamente desde su disco duro (vía red) sin necesidad de volcar el contenido a tu ordenador ...



El servidor que yo he montado para este proyecto tiene varios discos duros esclavos que serán usados como almacén que podrá compartirse con toda la red local, estos los he particionado desde la terminal con **fdisk**. Si necesitas ayuda para poder particionar tus discos duros desde la línea de comandos visita el tutorial ([Como particionar y formatear tu nuevo disco duro desde la línea de comandos en Linux Debian y Linux Ubuntu](#)) seguro que puede serte de ayuda.

Para comenzar como siempre no está demás decir que arranquemos nuestro **#Servidor Local** y nos identifiquemos como súper usuario (*root*) ...

```
Starting NFS common utilities: statd.
Starting portmap daemon...Already running..
Starting enhanced syslogd: rsyslogd.
Starting VirtualBox AdditionsVBoxService: 3.2.10_OSE r66523 started. Verbose level = 0
.
Starting ACPI services...
Starting deferred execution scheduler: atd.
Starting periodic command scheduler: cron.
Starting MTA: exim4.
Starting OpenBSD Secure Shell server: sshd.

Debian GNU/Linux 6.0 UFO tty1

UFO login: root
Password:
Linux UFO 2.6.32-5-686 #1 SMP Fri Sep 9 20:51:05 UTC 2011 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@UFO:~#
```

Instalación y configuración del servidor NFS ...

```
0000110001000100100101 CODIGO 1000100100000101000010
```

apt-get install portmap nfs-kernel-server

Una vez instalados vamos a editar el archivo de configuración (exports) con ..

```
0000110001000100100101 CODIGO 1000100100000101000010
```

vi /etc/exports

Agregamos el directorio que queremos compartir seguido de la dirección broadcast de nuestra red y los permisos así ...

```
0000110001000100100101 CODIGO 1000100100000101000010
```

/directorio/a/compartir 192.168.1.0/255.255.255.0(rw)

Si queremos mas directorios (o discos duros) podemos agregar tantas lineas como necesitemos. Una vez agregados nuestros directorios a compartir, pulsamos la tecla (ESC) y seguidamente (:wq!).

Damos permisos al directorio a compartir con ...

```
0000110001000100100101 CODIGO 1000100100000101000010
```

chmod -R 777 /directorio/a/compartir/

Reiniciamos los servicios para que coja los nuevos ajustes con ...

```
0000110001000100100101 CODIGO 1000100100000101000010
```

/etc/init.d/nfs-kernel-server restart

Y listo !! Ahora tan solo tenemos que configurar los ordenadores cliente para que puedan ver el directorio **compartido** por **NFS** de nuestro **#Servidor Local**. Para montar la **unidad compartida NFS** de nuestro servidor en unos de nuestros ordenadores con **Debian** o **Ubuntu** tan solo tendremos que teclear desde una terminal ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

```
sudo mount -t nfs IP.DE.TU.SERVER:/DIRECTORIO/COMPARTIDO
/DIRECTORIO/LOCAL/DONDE/MONTAR/LO/COMPARTIDO
```

Si quieres mas información puedes visitar el artículo ([Como instalar y configurar un servidor NFS y sus clientes en Linux Ubuntu y Linux Debian](#))

Seguridad en nuestro servidor NFS ...

Seguidamente y para terminar vamos a darle algo mas de seguridad a nuestro **servidor NFS** el cual comparte contenidos. Es cierto que comparte sus unidades con nuestra red local pero para estar seguros de que solo accedemos nosotros lo mejor será decirle que rango de IP es el que hay que aceptar y cual no. Para comenzar vamos a decirle que rango de IP queremos que (Si) sea admitido editando el siguiente archivo ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

```
vi /etc/hosts.allow
```

Bajamos al final del archivo e insertaremos las siguientes líneas ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

```
portmap:192.168.1.0/255.255.255.0
lockd:192.168.1.0/255.255.255.0
mountd:192.168.1.0/255.255.255.0
rquotad:192.168.1.0/255.255.255.0
statd:192.168.1.0/255.255.255.0
```

Como podéis ver estas IP y máscara de subred son las Broadcast de mi red. Estas debéis sustituirlas por vuestro rango. Salimos del editor pulsando la tecla (*ESC*) y seguidamente (*:wq!*).

Ahora vamos a denegar todas las direcciones IP que quieran acceder al servicio y no estén dentro de nuestro rango como le indicamos anteriormente en el fichero (*hosts.allow*) del siguiente modo ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

```
vi /etc/hosts.deny
```

Bajamos al final del fichero e insertamos lo siguiente ...

```
00001100010001001001001 CODIGO 10001001000001010100010
```

```
portmap:ALL
lockd:ALL
mountd:ALL
rquotad:ALL
statd:ALL
```

Salimos del editor como siempre pulsando la tecla (*ESC*) y seguidamente (*:wq!*). Una vez agregadas estas líneas ya lo tenemos mínimamente asegurado y podemos dormir tranquilos con nuestro servidor compartiendo cosas solo con nosotros y nadie más.

Vol 6 - Gestor de descargas vía web

Como no nuestro proyecto de **#Servidor Local** ([Viejos ordenadores que hacen grandes cosas](#)) tenia que tener un **gestor de descargas vía web**. Ya que lo tenemos encendido lo suyo seria que incorporará una web donde podamos decirle lo que queremos que nos descargue y listo. Imaginemos que estamos en casa de un amigo navegando por la red y nos gustaría descargarnos algo. En principio este tutorial explica como instalar el **gestor de descargas pyLoad** pero si [redireccionamos puertos en el Router](#) podemos manejarlo también remotamente y decirle desde casa de nuestro amigo que se ponga a descargar lo que queramos.



Este **gestor de descargas vía web** no solo es valido para descargas directas si no que es compatible con los Host ... *filesonic, megaupload, fileservr, wupload, uploaded* y *muchos mas*. Si te apetece usar tu servidor también para esta función arranca tu maquina, identificate como súper usuario (*root*) y ponte manos a la obra tecleando lo siguiente ...

```
0000110001001100100101 CODIGO 10001001000001010100010
```

```
wget http://get.pyload.org/get/ubuntu/pyload-v0.4.8-all.deb
```

Ahora lo instalaremos con ...

```
0000110001001100100101 CODIGO 10001001000001010100010
```

```
dpkg -i pyload-v0.4.8-all.deb
```

La instalación nos da problemas simplemente y avisa de que le faltan dependencias por cumplir para poder funcionar correctamente. Las instalaremos tecleando lo siguiente ...

```
0000110001001100100101 CODIGO 10001001000001010100010
```

```
apt-get -f install
```

Una vez instaladas las dependencias tecleamos ...

```
0000110001001100100101 CODIGO 10001001000001010100010
```

```
dpkg-reconfigure pyload
```

Ahora necesitamos crear un usuario que tenga permisos para poder acceder al **gestor de descargas vía web** y administrarlo. Para agregar uno o varios usuarios teclearemos lo siguiente ...



```
pyLoadCore -u
```

El programa nos dará varias opciones ...

```
Select action
1 - Create/Edit user
2 - List users
3 - Remove user
4 - Quit
[1]/2/3/4: 1
```

Nosotros vamos a crear un nuevo usuario pulsando (*1*) y la tecla (*Enter*) ...

```
Username [User]: nuevousuario
```

Insertamos un nombre de usuario y pulsamos la tecla (*Enter*) ...

```
Password : contraseña
```

Insertamos una contraseña para el nuevo usuario ...

```
Password ( again ): contraseña
```

Confirmamos nuestra nueva contraseña y pulsamos la tecla (*Enter*) ...

```
Select action
1 - Create/Edit user
2 - List users
3 - Remove user
4 - Quit
[1]/2/3/4: 4
```

Finalmente salimos del programa pulsando (*4*) y la tecla (*Enter*).

Ahora arrancaremos el **gestor de descargas PyLoad** con lo siguiente ...

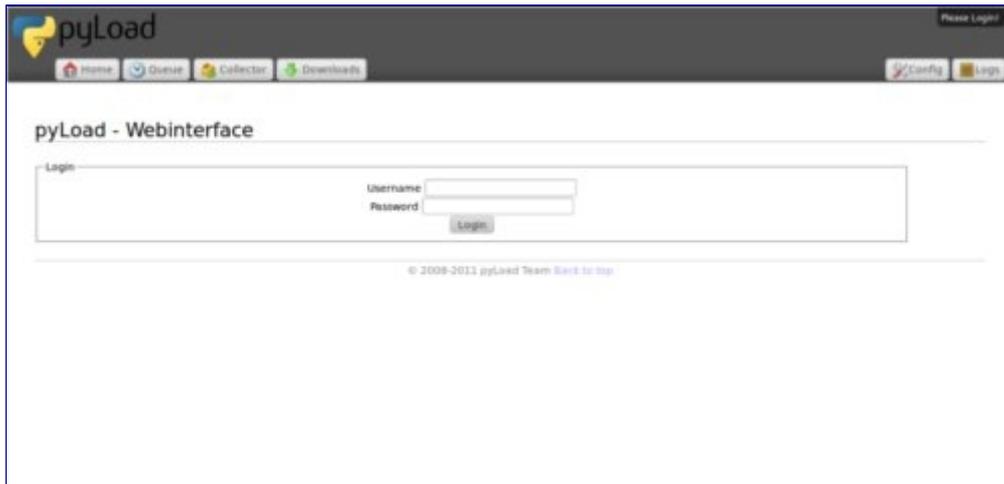


```
pyLoadCore
```

Ahora vámonos a otro ordenador de nuestra red e insertamos la *URL*

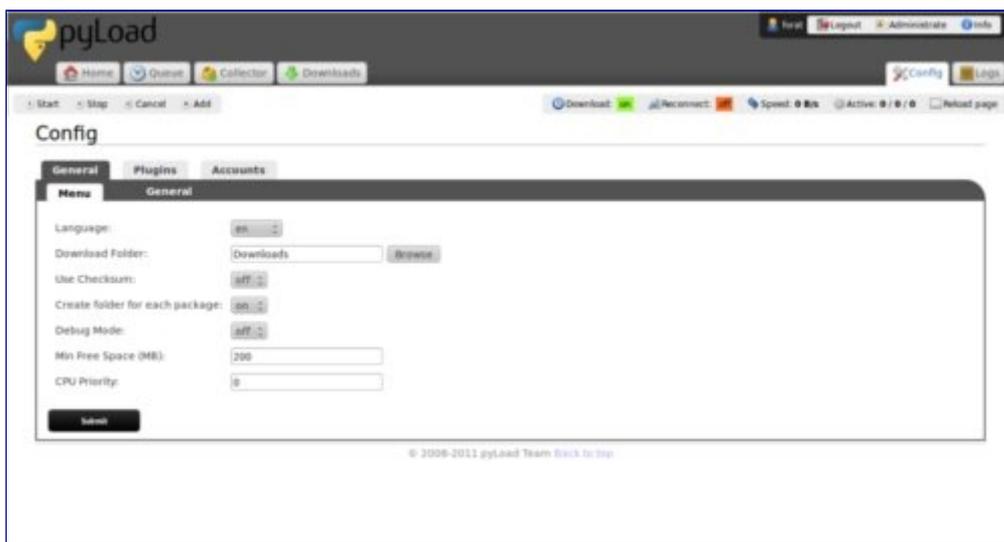
http://IP.DE.NUESTRO.SERVER:8001 (o *8000* dependiendo de la versión) y veremos lo siguiente

...



Insertamos nuestro nombre de usuario y contraseña que configuramos anteriormente y listo !

Una vez dentro y si lo necesitáis podéis cambiar el idioma desde la opción (*Config*), seguidamente situamos el ratón en la opción (*Menu*) y cuando aparezca el desplegable (*General*) ...



Hacemos click sobre la pestaña (*en*) y seleccionamos nuestro idioma. Una vez hecho esto y ya que **compartimos unidades por NFS** en nuestra red podemos aprovechar el momento para indicarle en la sección (*Download Folder*) el directorio que tenemos compartido. De este modo puedes acceder a los ficheros que descargues directamente en tu ordenador cliente por la red en vez de descargarlo al cliente y después ejecutarlo.

Para grabar nuestros cambios pulsamos el botón (*Submit*) y listo. Os invito a daros un paseo por la opciones de configuración del **gestor de descargas vía web** llamado **PyLoad** porque encontrareis opciones para hacer casi de todo con este programa.

Ahora volvemos a nuestro **#Servidor Local** y vamos a agregar la ruta del **gestor de descargas pyLoad** en el fichero (*rc.local*) para que se ejecute en el arranque de la maquina tecleando lo siguiente ...



vi /etc/rc.local

Justo antes de la línea (*exit 0*) insertamos lo siguiente ...



```
/usr/bin/pyLoadCore
```

Una vez insertada salimos del editor con la tecla (*ESC*) y seguidamente (*:wq!*). Ahora cuando reiniciemos se ejecutará automáticamente.

Recordemos que para agregar una descarga tan solo hay que pulsar sobre la opción (*+ add*) insertar un nombre, los enlaces que queramos descargar y el **gestor de descargas pyLoad** hará el resto. La descarga comenzará por momentos y podremos cerrar el navegador mientras que nuestro **#Servidor Local** hace su trabajo.

En el siguiente artículo montaremos un servidor de impresión para que podamos conectar nuestra impresora al servidor y poder imprimir desde todos los ordenadores clientes que tengamos conectados.

Vol 7 - Como montar un servidor de impresión

El que tiene varios ordenadores y una impresora suele ir intercambiando el cable USB de la impresora para imprimir desde el ordenador que necesite. Ya que vamos a tener nuestro *#Servidor Local* siempre conectado, podríamos instalarle a el la impresora y acceder desde todos los ordenadores que tengamos conectados a través de el. En el proyecto ([Viejos ordenadores que hacen grandes cosas](#)) vamos a encargarle esta tarea también a nuestro **viejo ordenador** ¿ *Quieres ver como ? ...*

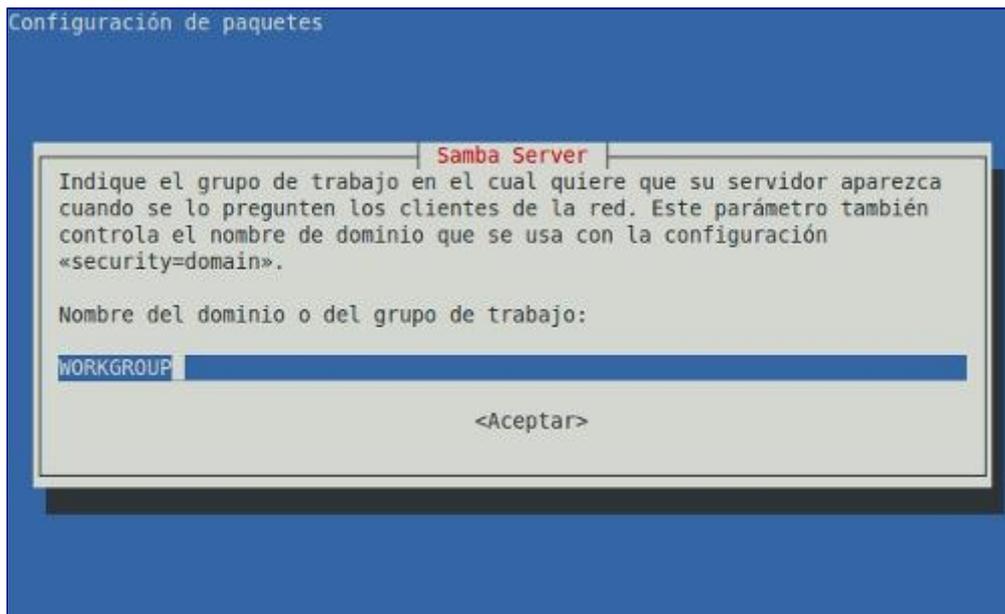


Instalación y configuración del servidor de impresión ...

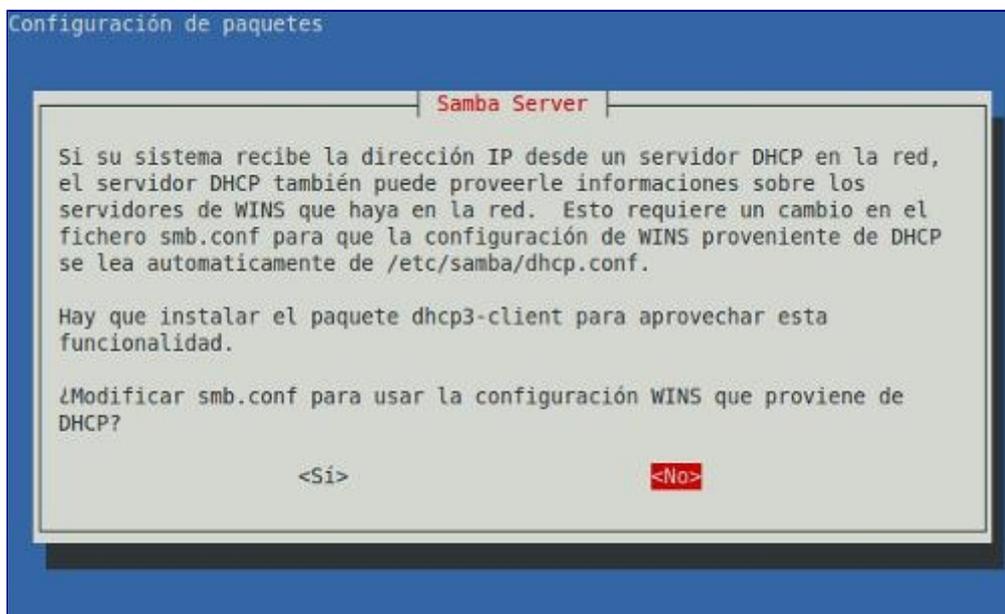
Como siempre arrancamos nuestro **servidor**, nos identificamos como súper usuario (*root*) y comenzamos por instalar el paquete (**cups**) tecleando lo siguiente ...



```
apt-get install cups
```



El sistema nos preguntará el grupo de trabajo de nuestra red. Podemos asignarle el nombre que queramos pero tendremos que indicárselo también a los ordenadores cliente desde los que queramos imprimir. El grupo de trabajo por defecto en todos los sistemas (*Windows*) es (*WORKGROUP*) con lo que si lo dejamos como está al menos los ordenadores con *Windows* que tengamos en la red no tendremos que configurarlos, por defecto al menos para clientes en **Ubuntu** no hay que configurar nada. Continuamos la instalación pulsando la tecla (*Enter*) ...



En este caso le indicaremos que no simplemente porque nuestro **servidor** tiene una **IP fija** que configuramos en el artículo ([Switch de red](#)). Continuamos pulsando la tecla (*Enter*) y finalizará su instalación.

Ahora vamos a editar su archivo de configuración y vamos a cambiar algunos parámetros tecleando lo siguiente ...

```
00001100010001001001 CODIGO 10001001000001010100010
```

```
vi /etc/cups/cupsd.conf
```

Buscamos las líneas ...

```
00001100010001001001 CODIGO 10001001000001010100010
```

```
# Only listen for connections from the local machine.  
Listen localhost:631  
Listen /var/run/cups/cups.sock
```

y añadimos nuestra **IP fija** en el **servidor** quedándonos así ...

```
00001100010001001001 CODIGO 10001001000001010100010
```

```
# Only listen for connections from the local machine.  
Listen localhost:631  
Listen /var/run/cups/cups.sock  
Listen 192.168.1.240 #Aquí la IP de nuestro servidor
```

Buscamos la línea ...

```
00001100010001001001 CODIGO 10001001000001010100010
```

```
DefaultAuthType Basic
```

Y le añadimos los siguiente ...

```
00001100010001001001 CODIGO 10001001000001010100010
```

```
DefaultAuthType Basic  
DefaultEncryption IfRequested
```

Seguidamente buscamos las líneas ...

```
00001100010001001001 CODIGO 10001001000001010100010
```

```
# Restrict access to the server...
```

```
Order allow,deny
```

Y añadimos nuestro rango de IP, sin este dato los demás ordenadores que tengamos en la red no podrán ver la impresora, lo modificaremos quedándonos así ...

```
00001100010001001001 CODIGO 10001001000001010100010
```

```
# Restrict access to the server...
```

```
Order allow,deny  
Allow localhost  
Allow 192.168.1.*
```

También deberemos decirle que dirección IP tiene el ordenador que puede administrar el **servidor CUPS**. Todos los ordenadores podrán acceder a este panel de control pero solo uno o los que queráis tendrán acceso a las secciones con permisos solo para el administrador del servidor ...



```
# Restrict access to the admin pages...
```

```
Order allow,deny
```

Añadimos nuestro servidor como localhost y la dirección IP del cliente administrador quedándonos así ...



```
# Restrict access to the admin pages...
```

```
Order allow,deny
Allow localhost
Allow 192.168.1.102
```

Al igual que en el caso anterior esta sección también necesita la IP del cliente administrador, así que busquemos las líneas ...



```
# Restrict access to configuration files...
```

```
AuthType Default
Require user @SYSTEM
Order allow,deny
```

Y la añadimos al final quedándonos así ...



```
# Restrict access to configuration files...
```

```
AuthType Default
Require user @SYSTEM
Order allow,deny
Allow localhost
Allow 192.168.1.102
```

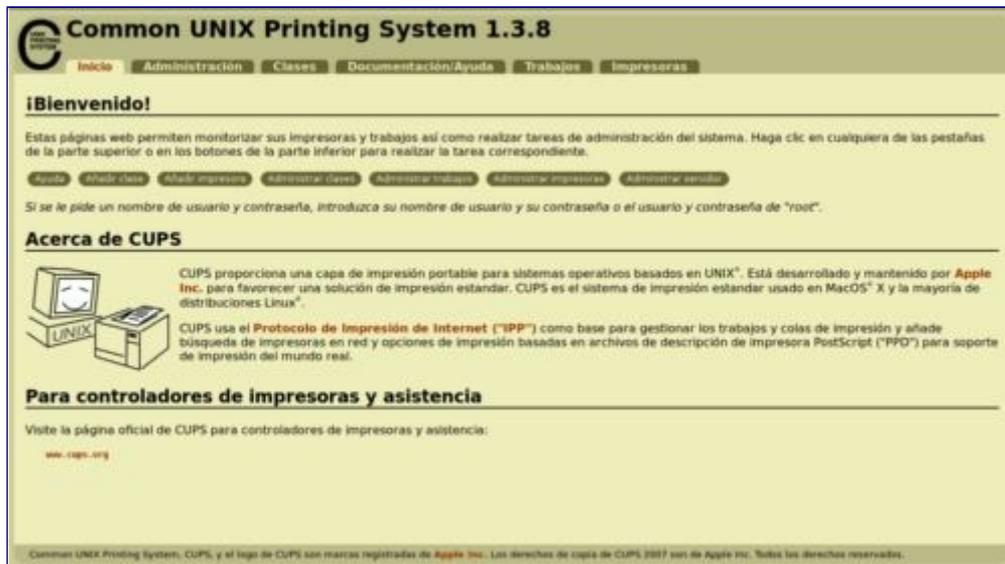
Una vez hechos los cambios salimos y guardamos el fichero de configuración pulsando la tecla (*ESC*) y seguidamente (*:wq!*).

Ahora reiniciaremos el **servidor de impresión cups** para que los cambios surjan efecto así ...



```
/etc/init.d/cups restart
```

Ahora abrimos un navegador en unos de nuestros ordenadores cliente e introducimos la IP de nuestro **servidor** seguido del puerto que **CUPS** reserva para el, por ejemplo (*http://IP.DE.NUESTRO.SERVER:631*) ...

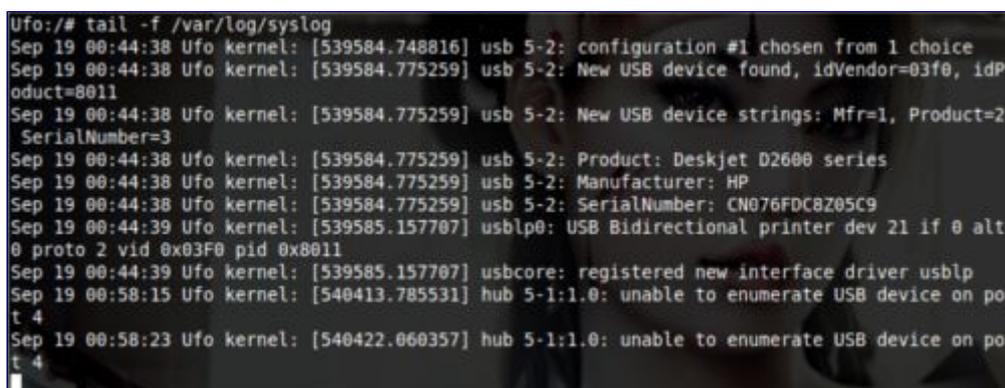


Antes de comenzar a imprimir cosas deberemos añadir una impresora que ya esté conectada al servidor. Antes de conectarla vamos a lanzar el comando (*tail*) sobre el fichero que almacena los informes del sistema. Con la impresora desconectada tecleamos ...



```
tail -f /var/log/syslog
```

Una vez lanzado el sistema se queda en espera, ahora conecta la impresora al servidor y deberíamos ver algo como esto ...



Mi impresora *HP Deskjet D2660* ha sido detectada como una impresora *HP Deskjet* y se le ha asignado el driver de la serie *D2600*. Esta impresora me la he comprado este año por lo que es algo actual, a estas alturas seguramente la tuya también será detectada automáticamente, de no ser así pon tu modelo de impresora en *Google* seguido de (**Linux Debian**) y seguramente encontrarás

info. Seguimos con el comando *tail* que esta ahí esperando a ser abortado. Pulsamos simultáneamente las teclas (*Ctrl + z*) y listo.

Volvemos a nuestro navegador donde teníamos al **servidor de impresión CUPS**, pulsamos sobre la opción (*Añadir impresora*) ...

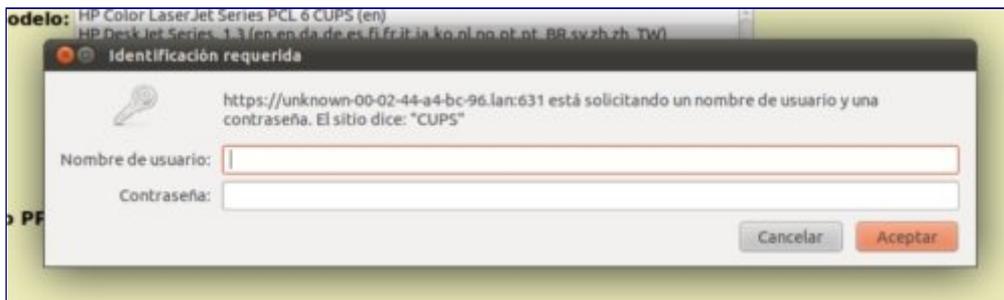
Introducimos en el apartado (*Nombre*) un nombre para nuestra impresora como por ejemplo (*HP*), en el apartado (*Ubicación*) dejamos por ejemplo (*Casa*) y en descripción (*HP Deskjet D2660*). Una vez rellenados los apartados continuamos pulsando sobre el botón (*Siguiente*) ...

Como podemos ver seleccionamos el *Driver* con el que nos detecto la impresora nuestro sistema, una vez marcado continuamos pulsando sobre el botón (*Siguiente*) ...



Seleccionamos el modelo que mas se asemeje a nuestra impresora o introducimos un archivo *ppd* que es un *driver* para que detecte tu impresora. Si no te detecta la impresora prueba a buscar tu archivo *ppd* en el [buscador de CUPS](#).

Continuamos pulsando sobre el botón (*Siguiente*) ...



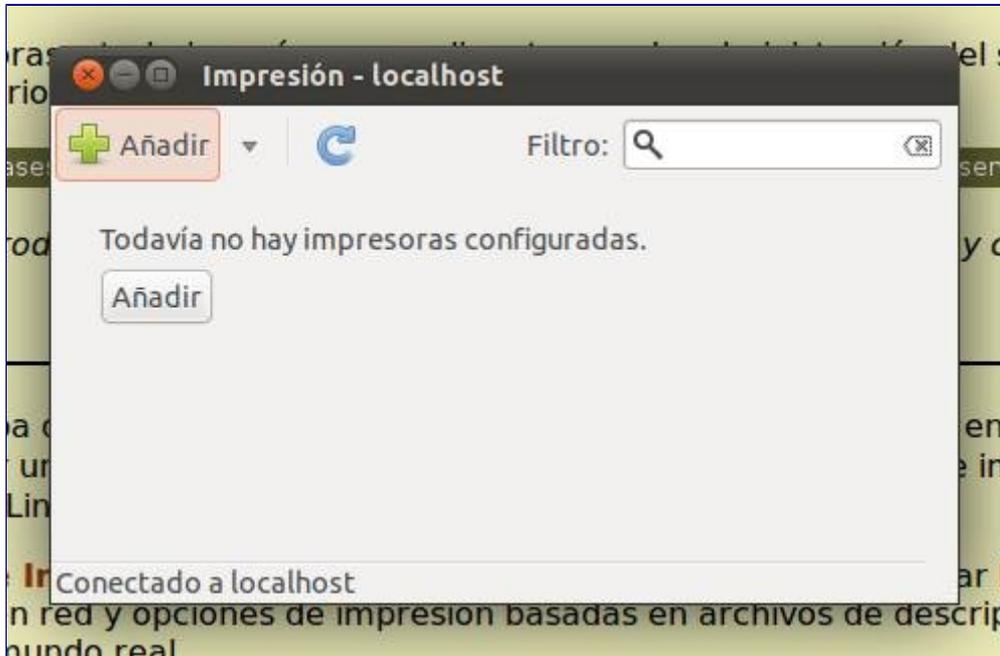
Introducimos el nombre de usuario (*root*) y nuestra contraseña de administrador en el sistema, seguidamente pulsamos la tecla (*Enter*) ...



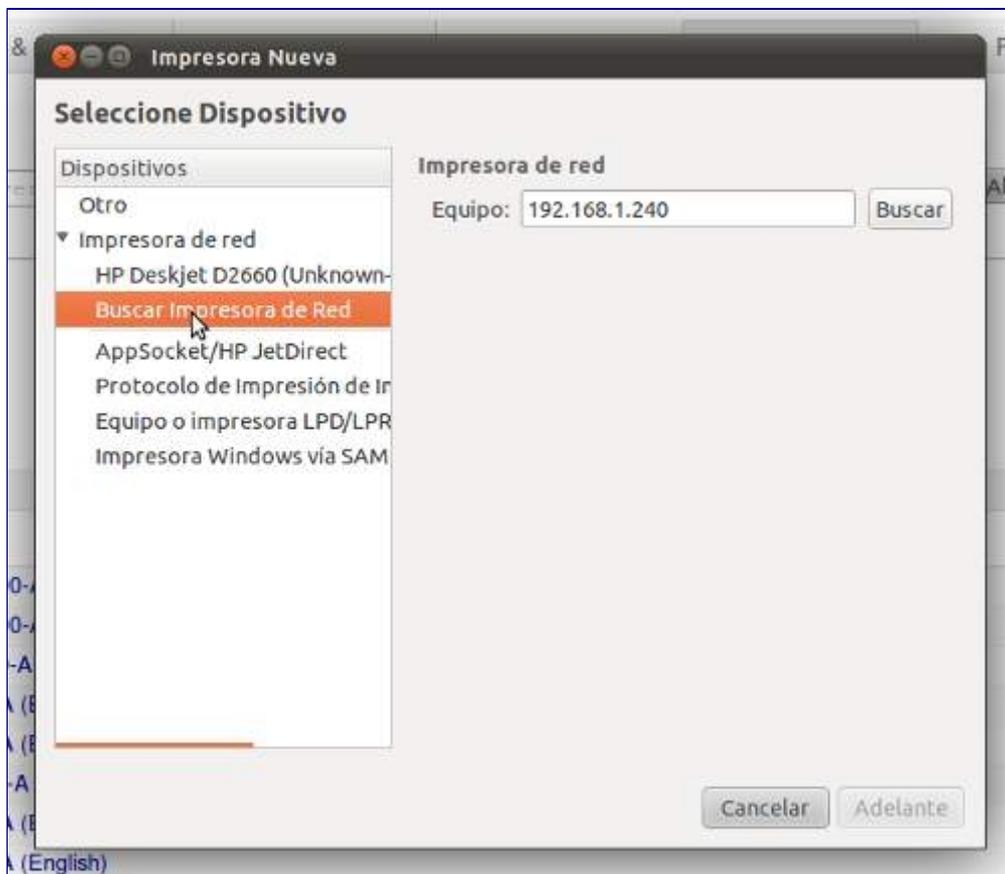
En este apartado podemos configurar la calidad de la impresión, color, papel, resolución, etc ... Ahora tan solo tenemos que agregar nuestra nueva impresora en red en los ordenadores cliente.

Instalación y configuración del cliente ...

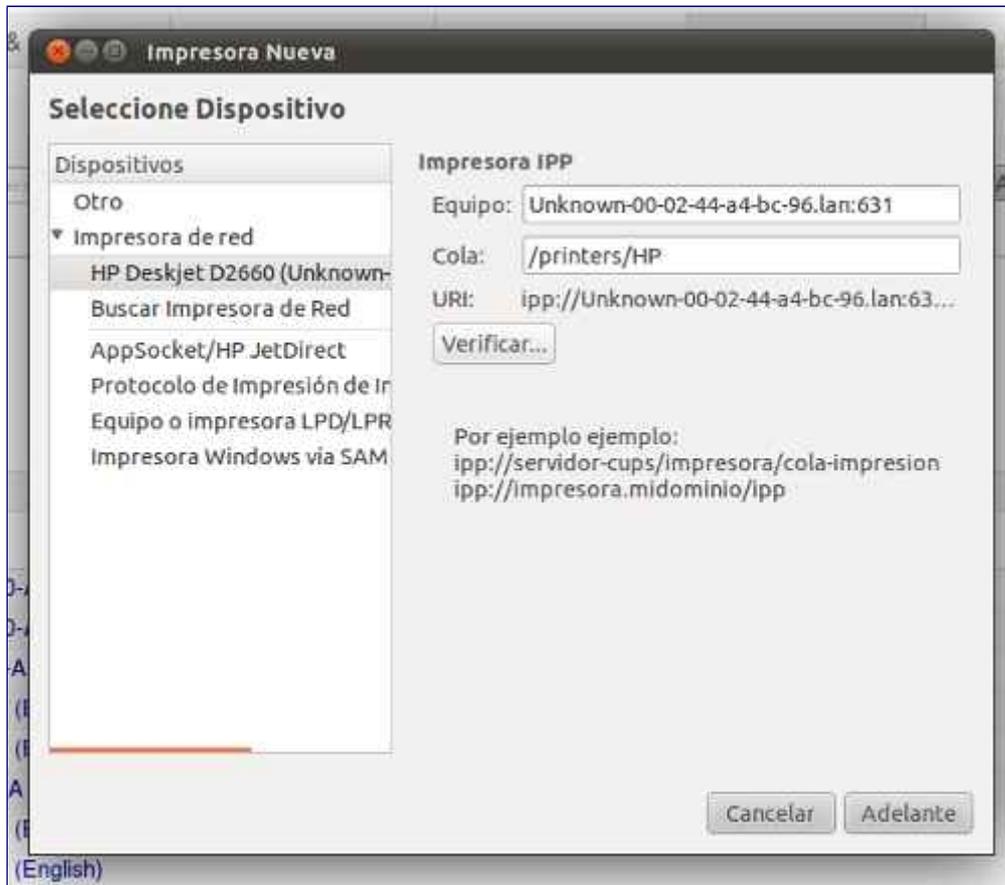
En mi caso el ordenador cliente desde el que me conecto a mi impresora en red tiene instalado **Linux Ubuntu 11.04**. Para agregar una impresora tan solo tenemos que acceder al menú *Aplicaciones -> Sistema -> Impresión ...*



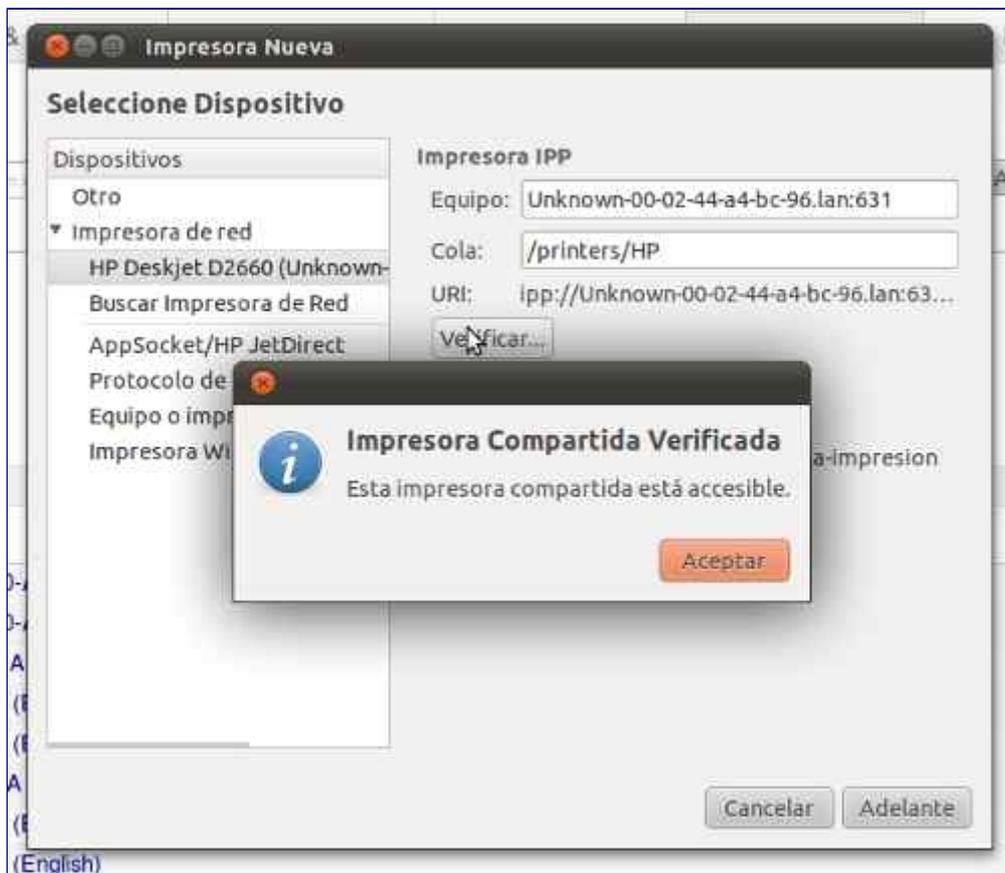
Pulsamos el botón (*Añadir*) ...



Seleccionamos la opción (*Impresora de red -> Buscar impresora de red*), introducimos la **IP** de nuestro **#Servidor Local** en la casilla y pulsamos el botón (*Buscar*) ...



Como podemos ver aquí tenemos la impresora que esta conectada a nuestro servidor. Pulsamos sobre el botón (*Verificar*) ...



Si no os sale este mensaje, cambiad la dirección Unknown-00-01.bla.bla por la dirección IP de vuestro servidor seguido de el puerto como por ejemplo 192.168.1.240:631

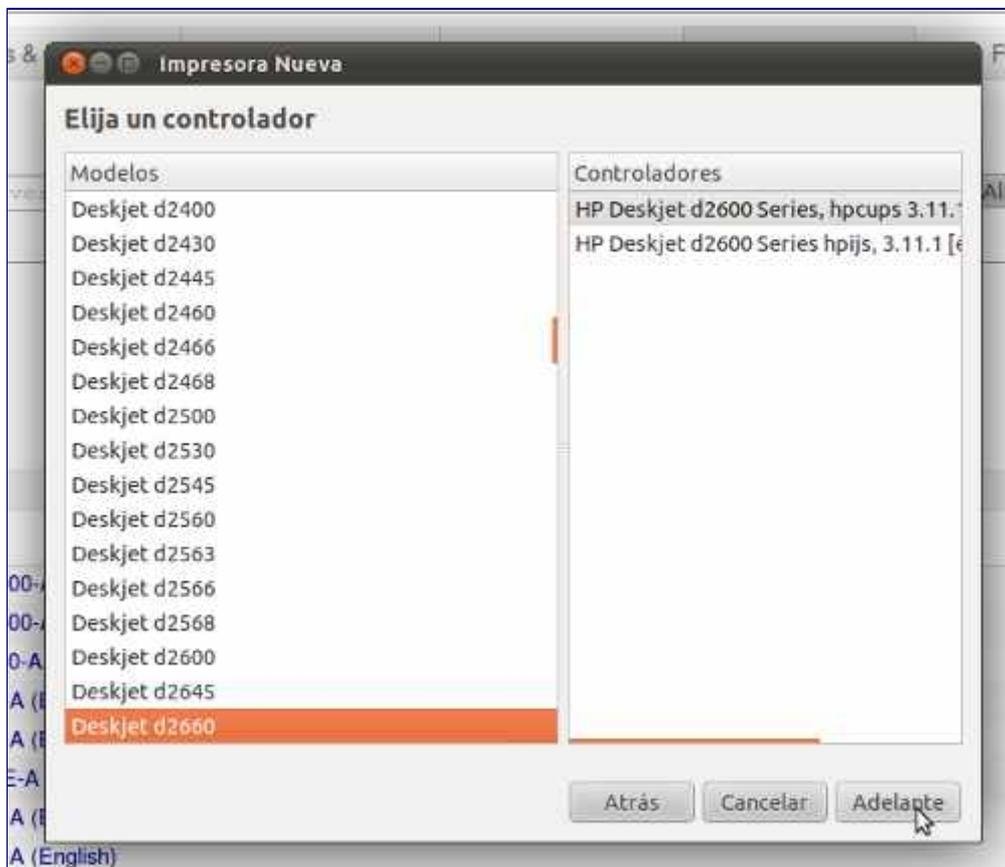
Pulsamos la tecla (*Enter*) y pulsamos el botón (*Adelante*) ...



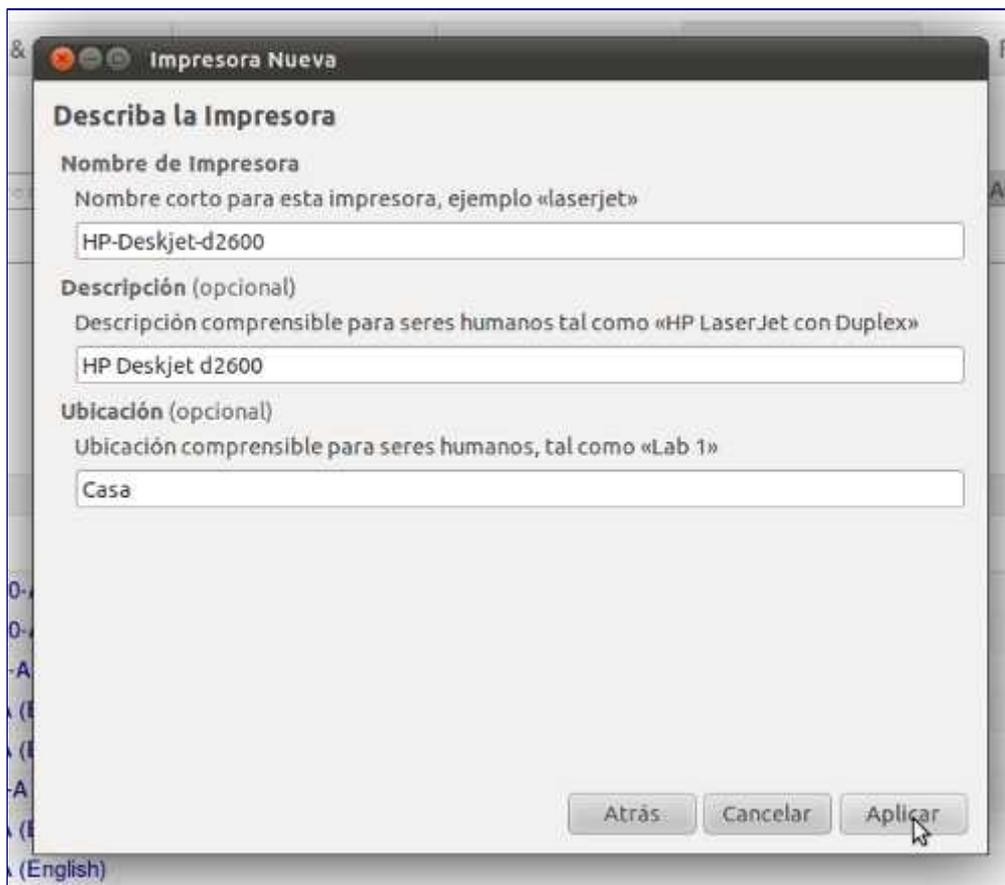
El sistema necesita los drivers de nuestra impresora remota para poder usarla ...



Seleccionamos el fabricante de nuestra impresora y continuamos pulsando el botón (*Adelante*) ...



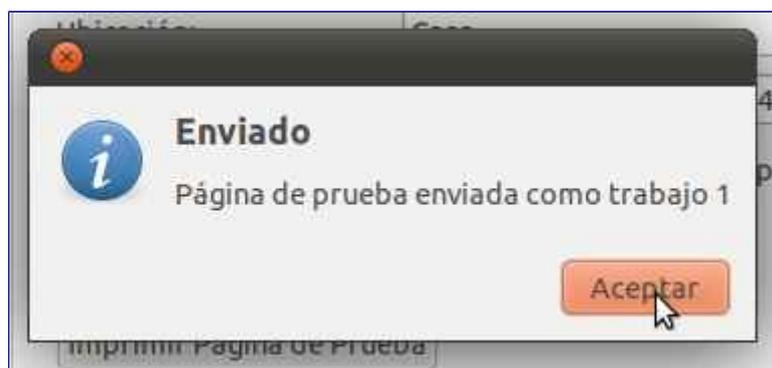
Seleccionamos nuestro modelo, el driver que queremos usar y pulsamos el botón (*Adelante*) ...



La configuramos en el **servidor** y ahora podemos cambiar estos parámetros aunque no es necesario, continuamos pulsando el botón (*Aplicar*) ...



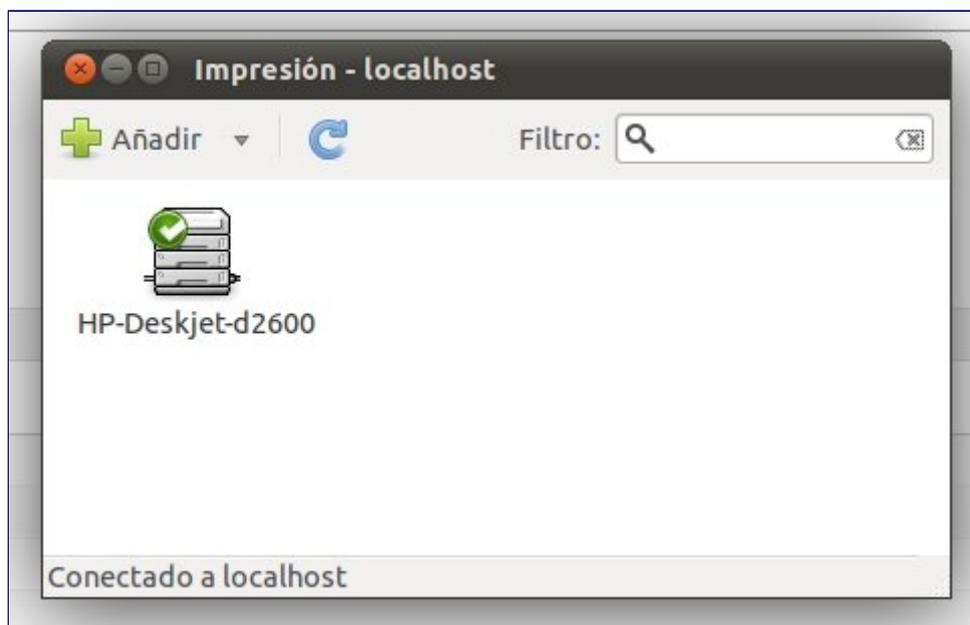
El sistema nos propone imprimir una pagina de prueba, (*porque no !!*) vamos a probar a ver que tal funciona nuestra **impresora en red** ...



Las tarjetas de red se ponen manos a la obra y se pasan la información para que se ponga a imprimir en un periquete, efectivamente la pagina de prueba se acaba de imprimir ...



Es una lastima pero no puedo indicaros como configurar vuestras impresoras en todos los sistemas operativos que useis. Mas o menos todos vienen necesitando lo mismo pero siguen otros pasos. Tendreis que usar vuestra destreza para configurar la impresora en vuestros clientes. Si el servidor os la detecta y habeis sido capaces de llegar a configurar la calidad de impresión y todo lo demás es señal de que todo va bien ...



Si es así, tendréis que centrar mas atención sobre el cliente ya que el error posiblemente puede estar en el, *la forma de agregar una impresora en red, configuración, drivers usados, etc ...*

Este tutorial dedicado a montar un **servidor de impresión CUPS** esta basado y probado bajo **Debian GNU/Linux** instalado en el servidor y **Linux Ubuntu 11.04** en el cliente. Funciona a la perfección tambien bajo **Ubuntu 11.10** aunque no se a ciencia cierta si en otros sistemas funciona sin problemas.

Vol 8 - Web de control y phpSysInfo

Para terminar he pensado en montar una pequeña web donde tengamos acceso directo a lo que se puede acceder desde el navegador como puede ser el [gestor de descargas pyLoad](#), el [servidor de impresión CUPS](#), el administrador de las [bases de datos phpMyadmin](#) o las constantes vitales de nuestro [servidor con phpSysInfo](#). No es que sea una gran hazaña montar una *mini-web* con una imagen y cuatro enlaces pero he pensado que no estaría mal para que todo el mundo pueda situarse en su servidor a golpe de click.

Como ultimo agregado a nuestro querido #Servidor Local no es otro programa que **phpSysInfo** el cual nos ofrece información instantánea vía web sobre el estado del [Hardware](#) de la maquina ...



Si os animáis a instalar este programa su instalación es tan sencilla como conectarnos a la maquina como súper usuario (*root*) y tecleamos lo siguiente ...

```
000010001001001001001001 CODIGO 10001001000001010100010
```

```
apt-get install phpsysinfo
```

Y crear un enlace simbólico al directorio web de nuestro **servidor** con ...

```
000010001001001001001001 CODIGO 10001001000001010100010
```

```
ln -s /usr/share/phpsysinfo /home/usuario/web/phpsysinfo
```

Agregado este ultimo vamos a montar la *mini-web* que tendremos que alojar en un directorio dentro de la raíz de nuestro **servidor**, si has seguido los tutoriales de la saga ([Viejos ordenadores que hacen grandes cosas](#)) tendrás que crear el directorio dentro de (*/home/usuario/web/*) con ...

```
000010001001001001001001 CODIGO 10001001000001010100010
```

```
cd /home/usuario/web
```

Creamos el directorio ...

```
000010001001001001001001 CODIGO 10001001000001010100010
```

```
mkdir webcontrol
```

Accedemos al nuevo directorio ...

```
0000110001000100100100101 CODIGO 1000100100000101000010
```

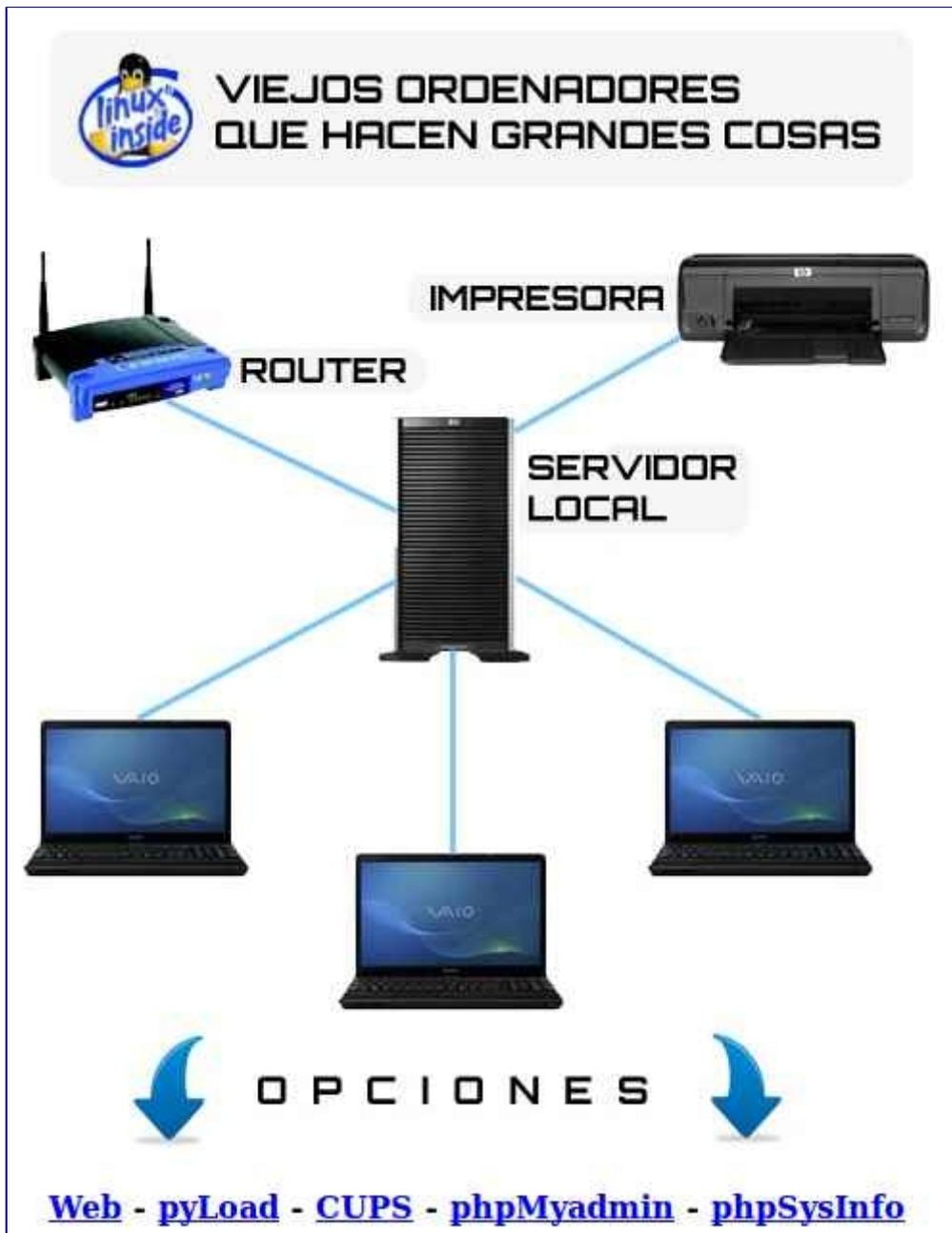
```
cd webcontrol
```

Ahora nos descargamos el fichero HTML y la imagen así ...

```
0000110001000100100100101 CODIGO 1000100100000101000010
```

```
wget http://dl.dropbox.com/u/3252559/webcontrol/index.html  
wget http://dl.dropbox.com/u/3252559/webcontrol/vie.jpg
```

Una vez hecho esto vamos a un ordenador cliente, arrancamos un navegador e introducimos la IP de tu servidor seguido del directorio que acabamos de crear, por *ej**
http://IP.DE.TU.SERVER/webcontrol y veremos lo siguiente ...



Como podemos ver ahí está la imagen y el fichero HTML que descargamos fusionado en una especie de web. Esta tan solo trae con ella enlaces hacia nuestros servicios. Si haces click en los enlaces veras que no te funcionan porque no apuntan hacia tu servidor.

Para modificar esto vamos a volver a la terminal de nuestro **#Servidor Local** y editaremos el fichero HTML con ...



vi /home/usuario/web/webcontrol/index.html

Veremos lo siguiente ...

```
<html><head>
<meta http-equiv="content-type" content="text/html; charset=ISO-8859-1">

<title>Web Control</title>

</head><body>
<table bgcolor="#FFFFFF" height="100%" width="100%">
<tbody><tr>
<td align="center" valign="center">

</td>
</tr>
<tr>
<td align="center" valign="center">

<strong><a href="http://IP.DE.TU.SERVER">Web</a> -
<a href="http://IP.DE.TU.SERVER:8000/">pyLoad</a> - <a href="http://
IP.DE.TU.SERVER:631">CUPS</a> - <a href="http://IP.DE.TU.SERVER/
phpmyadmin">phpMyadmin</a> - <a href="http://IP.DE.TU.SERVER/
phpsysinfo">phpSysInfo</a></strong>

</td>
</tr>
</tbody></table>
</body></html>
```

Si no has cambiado las rutas de los programas que hemos instalado durante todo el proyecto tan solo tendrás que cambiar (*IP.DE.TU.SERVER*) por la *IP de tu servidor* y listo. *De ahí la sencillez de la web ...*

Una vez introducida nuestra IP en todos los enlaces salimos del editor pulsando la tecla (*ESC*) y seguidamente (*:wq!*). Volvemos de nuevo al navegador de un ordenador cliente y cargamos de nuevo la web de control con (*http://IP.DE.TU.SERVER/webcontrol*) y ahora ya podemos acceder a lo que nos ofrece el servidor vía web 😊

Sois libres de copiar, modificar o incluso usar este manual para lo que queráis
incluidos los estudiantes universitarios como proyecto para presentarlo.

Este manual ha sido creado para el aprendizaje de todo el que le interese y
puede hacer con el lo que quiera, tan solo se pide un reconocimiento del mismo
con un link en tu Web